CWE_Analysis

December 6, 2020

1 Group 9

2 Saba Janamian, Yuping Yu

3 DSE 203 Final Project - Common Vulnerability and Exposure Analysis

[78]:

```
import os
import json
from neo4j import GraphDatabase
import codecs
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set_theme(style="whitegrid")
```

4 Connecting to Neo4j API

[79]:

```
uri = "neo4j://localhost:7687"
userName = "neo4j"
password = "password"
```

[80]:

```
# Connect to the neo4j database server
graph_db_driver  = GraphDatabase.driver(uri, auth=(userName, password))
```

[81]:

```
base_dir = '/Users/janamian/Documents/workstation/ucsd_dse_program/fall_2019/
 ↪docker_vol/saba-ja/workstation/dse_203_2020/project/
 ↪dse_203_final_project_fall_2020/data'
```

[82]:

```
with open(os.path.join(base_dir, 'cwe_data', 'cwec_v4.2.json')) as f:
    cwe = json.load(f)

# with open(os.path.join(base_dir, 'nvd_data', 'nvdcve-1.1-2020.json')) as f:
#     nvd = json.load(f)
```

5 Count types

[83]:

```
cwe_type_counter = {}
for obj in cwe['Weakness_Catalog']['Weaknesses']['Weakness']:
    for key, val in obj.items():
        t = cwe_type_counter.get(key, 0)
        if t == 0:
            cwe_type_counter[key] = 1
        else:
            cwe_type_counter[key] = cwe_type_counter[key] + 1
```

[84]:

```
cwe_type_counter
```

[84]:

```
{'@ID': 914,
 '@Name': 914,
 '@Abstraction': 914,
 '@Structure': 914,
 '@Status': 914,
 'Description': 914,
 'Extended_Description': 650,
 'Related_Weaknesses': 882,
 'Applicable_Platforms': 666,
 'Background_Details': 44,
 'Modes_Of_Introduction': 776,
 'Likelihood_Of_Exploit': 187,
 'Common_Consequences': 870,
 'Potential_Mitigations': 614,
 'Demonstrative_Examples': 475,
 'Observed_Examples': 392,
 'References': 496,
 'Content_History': 914,
 'Weakness_Ordinalities': 256,
 'Alternate_Terms': 83,
 'Detection_Methods': 89,
 'Taxonomy_Mappings': 628,
 'Related_Attack_Patterns': 273,
 'Notes': 313,
 'Affected_Resources': 51,
 'Functional_Areas': 24}
```

[85]:

```
data = {'attribute_element':[], 'count':[]}
for key, value in cwe_type_counter.items():
#     if '@' not in key:
    data['attribute_element'].append(key)
    data['count'].append(value)

cwe_type_df = pd.DataFrame(data)
```

```
cwe_type_df.sort_values(by=['count'], ascending=False, inplace=True)
cwe_type_df.reset_index(drop=True, inplace=True)
```

[86]:

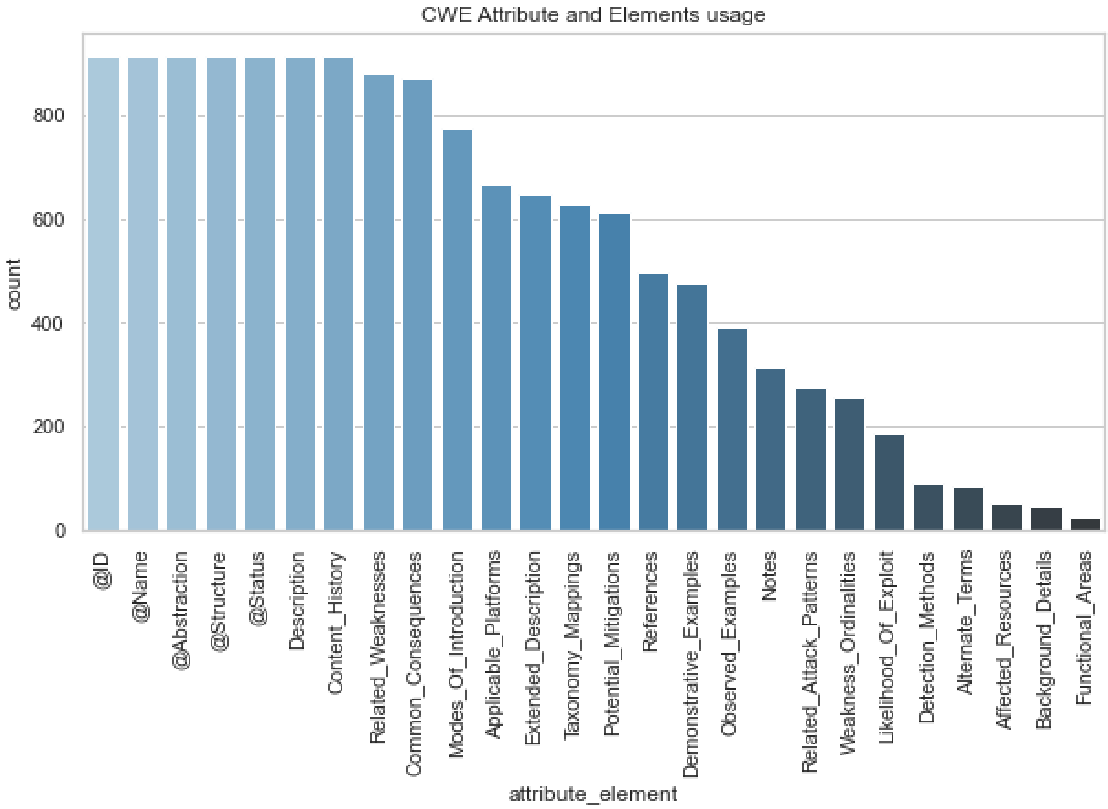
```
cwe_type_df
```

[86]:

```
         attribute_element  count
0                      @ID    914
1                    @Name    914
2             @Abstraction    914
3               @Structure    914
4                  @Status    914
5              Description    914
6          Content_History    914
7       Related_Weaknesses    882
8      Common_Consequences    870
9    Modes_Of_Introduction    776
10    Applicable_Platforms    666
11    Extended_Description    650
12       Taxonomy_Mappings    628
13   Potential_Mitigations    614
14              References    496
15  Demonstrative_Examples    475
16       Observed_Examples    392
17                   Notes    313
18 Related_Attack_Patterns    273
19   Weakness_Ordinalities    256
20   Likelihood_Of_Exploit    187
21       Detection_Methods     89
22         Alternate_Terms     83
23      Affected_Resources     51
24      Background_Details     44
25         Functional_Areas     24
```

[87]:

```
plt.figure(figsize=(10,5))
ax = sns.barplot(x="attribute_element", y="count", palette="Blues_d",␣
 ↪data=cwe_type_df);
plt.title('CWE Attribute and Elements usage')
plt.setp(ax.get_xticklabels(), rotation=90);
```



[88]:

```
def run_query(q):
    with graph_db_driver.session() as graph_db_session:
        try:
            graph_db_session.run(q)
        except:
            print(q)
            raise NameError
```

[89]:

```
def get_cwe_name_by_id(cwe_id_val):
    for obj in cwe['Weakness_Catalog']['Weaknesses']['Weakness']:
        cwe_id = obj['@ID']
        if int(cwe_id) == int(cwe_id_val):
            return obj['@Name']
```

6 Create CWE nodes and properties

[90]:

```
# ####################
# Create CWE nodes and properties
# ####################
delete_query = 'MATCH (n) DETACH DELETE n'
```

```
run_query(delete_query)
for obj in cwe['Weakness_Catalog']['Weaknesses']['Weakness']:
    cwe_id = obj['@ID']
    name = obj['@Name'].replace('\\','\\\\').replace('"', '\\"').replace("'",␣
↪"\\'")
    desc = obj['Description'].replace('\\','\\\\').replace('"', '\\"').
↪replace("'", "\\'")
    likelihood_of_exploit = obj.get('Likelihood_Of_Exploit', 'Unknown')
    if likelihood_of_exploit == "Unknown":
        community = 0
    elif likelihood_of_exploit == "Low":
        community = 1
    elif likelihood_of_exploit == "Medium":
        community = 2
    elif likelihood_of_exploit == "High":
        community = 3
    else:
        print(cwe_id, likelihood_of_exploit)

    cql_create_cwe_node = f"""CREATE (:cwe {{ cwe_id: "{cwe_id}",
    name: "{name}",
    description: "{desc}",
    community: {community},
    likelihood_of_exploit: "{likelihood_of_exploit}"
    }})"""
    run_query(cql_create_cwe_node)
```

[91]:
```
cql_create_cwe_node = f"""CREATE (:cwe {{ cwe_id: "NVD-CWE-Other",
    name: "Other",
    description: "NVD is only using a subset of CWE for mapping instead of the␣
↪entire CWE, and the weakness type is not covered by that subset.",
    community: 0,
    likelihood_of_exploit: "Unknown"
    }})"""
run_query(cql_create_cwe_node)
```

[92]:
```
cql_create_cwe_node = f"""CREATE (:cwe {{ cwe_id: "NVD-CWE-noinfo",
    name: "Insufficient Information",
    description: "There is insufficient information about the issue to classify␣
↪it; details are unkown or unspecified.",
    community: 0,
    likelihood_of_exploit: "Unknown"
    }})"""
run_query(cql_create_cwe_node)
```

[93]:

```
cql_create_cwe_node = f"""CREATE (:cwe {{ cwe_id: "NVD-no-analysis",
    name: "No Analysis",
    description: "CVEs mapping to this CWE are either rejected or do not have␣
 ↪any mapping to any CWE",
    community: 0,
    likelihood_of_exploit: "Unknown"
    }})"""
run_query(cql_create_cwe_node)
```

7 Create CWE weakness relationship

[94]:

```
# ####################
# Create CWE weakness relationship
# ####################
relationship_set = set()
relationship_count = {}
for obj in cwe['Weakness_Catalog']['Weaknesses']['Weakness']:
    cwe_id = obj['@ID']
    name = obj['@Name']
    try:
        rel_obj = obj['Related_Weaknesses']['Related_Weakness']
    except KeyError:
        print(f'{cwe_id} has no outward weakness relationship. Name: {name}')
        continue

    if not isinstance(rel_obj, list):
        rel_obj_list = [rel_obj]
    else:
        rel_obj_list = rel_obj

    for rel in rel_obj_list:
        related_cwe_id = rel['@CWE_ID']
        relationship = rel['@Nature']
        rel_str = f'({cwe_id})->[{relationship}]->({related_cwe_id})'
        if rel_str in relationship_set:
            continue
        else:
            relationship_set.add(rel_str)

        if relationship == 'ChildOf':
            child_count = relationship_count.get(related_cwe_id, 0)
            child_count += 1
            relationship_count[related_cwe_id] = child_count

        cql_create_relationship = f"""MATCH (cwe1:cwe),(cwe2:cwe)
```

```
                                      WHERE cwe1.cwe_id = '{cwe_id}' AND cwe2.
 ↪cwe_id = '{related_cwe_id}'
                                      CREATE (cwe1)-[r:{relationship}]->(cwe2)
                                      RETURN type(r)"""
        run_query(cql_create_relationship)

for key, val in relationship_count.items():
    cql_update_cwe_node = f"""MATCH (c:cwe {{ cwe_id: "{key}"}}) SET c.
 ↪child_count = {val}"""
    run_query(cql_update_cwe_node)
```

```
1187 has no outward weakness relationship. Name: DEPRECATED: Use of
Uninitialized Resource
132 has no outward weakness relationship. Name: DEPRECATED (Duplicate):
Miscalculated Null Termination
216 has no outward weakness relationship. Name: DEPRECATED: Containment Errors
(Container Errors)
217 has no outward weakness relationship. Name: DEPRECATED: Failure to Protect
Stored Data from Modification
218 has no outward weakness relationship. Name: DEPRECATED (Duplicate): Failure
to provide confidentiality for stored data
225 has no outward weakness relationship. Name: DEPRECATED (Duplicate): General
Information Management Problems
247 has no outward weakness relationship. Name: DEPRECATED (Duplicate): Reliance
on DNS Lookups in a Security Decision
249 has no outward weakness relationship. Name: DEPRECATED: Often Misused: Path
Manipulation
284 has no outward weakness relationship. Name: Improper Access Control
292 has no outward weakness relationship. Name: DEPRECATED (Duplicate): Trusting
Self-reported DNS Name
373 has no outward weakness relationship. Name: DEPRECATED: State
Synchronization Error
423 has no outward weakness relationship. Name: DEPRECATED (Duplicate): Proxied
Trusted Channel
435 has no outward weakness relationship. Name: Improper Interaction Between
Multiple Correctly-Behaving Entities
443 has no outward weakness relationship. Name: DEPRECATED (Duplicate): HTTP
response splitting
458 has no outward weakness relationship. Name: DEPRECATED: Incorrect
Initialization
516 has no outward weakness relationship. Name: DEPRECATED (Duplicate): Covert
Timing Channel
533 has no outward weakness relationship. Name: DEPRECATED: Information Exposure
Through Server Log Files
534 has no outward weakness relationship. Name: DEPRECATED: Information Exposure
Through Debug Log Files
542 has no outward weakness relationship. Name: DEPRECATED: Information Exposure
```

```
Through Cleanup Log Files
545 has no outward weakness relationship. Name: DEPRECATED: Use of Dynamic Class
Loading
592 has no outward weakness relationship. Name: DEPRECATED: Authentication
Bypass Issues
596 has no outward weakness relationship. Name: DEPRECATED: Incorrect Semantic
Object Comparison
664 has no outward weakness relationship. Name: Improper Control of a Resource
Through its Lifetime
691 has no outward weakness relationship. Name: Insufficient Control Flow
Management
693 has no outward weakness relationship. Name: Protection Mechanism Failure
697 has no outward weakness relationship. Name: Incorrect Comparison
703 has no outward weakness relationship. Name: Improper Check or Handling of
Exceptional Conditions
707 has no outward weakness relationship. Name: Improper Neutralization
71 has no outward weakness relationship. Name: DEPRECATED: Apple '.DS_Store'
710 has no outward weakness relationship. Name: Improper Adherence to Coding
Standards
769 has no outward weakness relationship. Name: DEPRECATED: Uncontrolled File
Descriptor Consumption
92 has no outward weakness relationship. Name: DEPRECATED: Improper Sanitization
of Custom Special Characters
```

[95]:

```python
data = {'parent_id':[], 'parent_name': [], 'child_count':[]}
for key, value in relationship_count.items():
    data['parent_id'].append(key)
    p_name = get_cwe_name_by_id(key).split(' ')[0:6]
    p_name = ' '.join(p_name)
    p_name = f'{p_name} ({key})'
    data['parent_name'].append(p_name)
    data['child_count'].append(value)

cwe_child_count_df = pd.DataFrame(data)
cwe_child_count_df.sort_values(by=['child_count'], ascending=False,␣
 ↪inplace=True)
cwe_child_count_df.reset_index(drop=True, inplace=True)
cwe_child_count_gt_n = cwe_child_count_df[cwe_child_count_df['child_count'] >=␣
 ↪20]
cwe_child_count_gt_n2 = cwe_child_count_df[cwe_child_count_df['child_count'] >=␣
 ↪10]
cwe_child_count_gt_n3 = cwe_child_count_df[cwe_child_count_df['child_count'] >=␣
 ↪5]
```

[96]:

```python
cwe_child_count_df
```

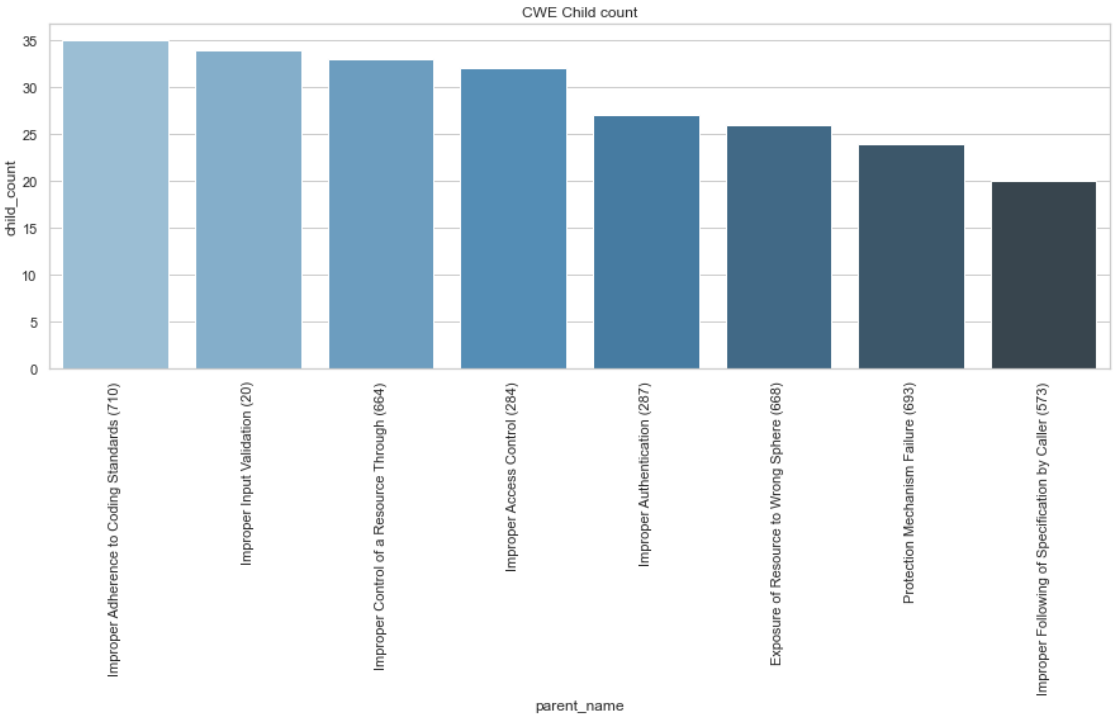
```
[96]:     parent_id                                         parent_name  child_count
     0          710        Improper Adherence to Coding Standards (710)           35
     1           20                       Improper Input Validation (20)           34
     2          664        Improper Control of a Resource Through (664)           33
     3          284                        Improper Access Control (284)           32
     4          287                        Improper Authentication (287)           27
     ..         …                                                  …          …
     225       1286  Improper Validation of Syntactic Correctness o…           1
     226        489                               Active Debug Code (489)            1
     227        524  Use of Cache Containing Sensitive Information …           1
     228       1229                  Creation of Emergent Resource (1229)            1
     229         96  Improper Neutralization of Directives in Stati…           1

     [230 rows x 3 columns]
```

```
[97]: cwe_child_count_gt_n
```

```
[97]:   parent_id                                         parent_name  child_count
     0        710        Improper Adherence to Coding Standards (710)           35
     1         20                       Improper Input Validation (20)           34
     2        664        Improper Control of a Resource Through (664)           33
     3        284                        Improper Access Control (284)           32
     4        287                        Improper Authentication (287)           27
     5        668          Exposure of Resource to Wrong Sphere (668)           26
     6        693                  Protection Mechanism Failure (693)           24
     7        573  Improper Following of Specification by Caller …           20
```

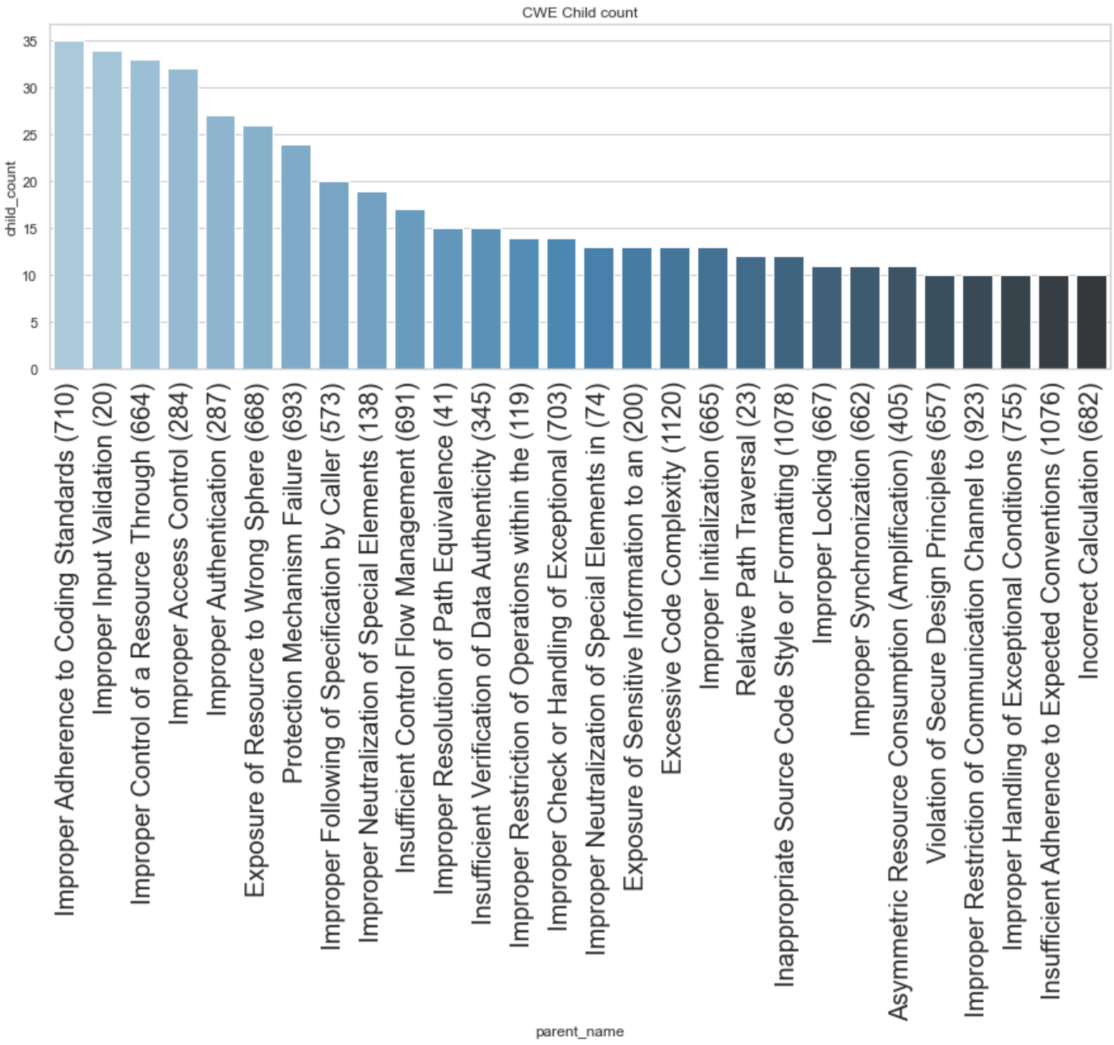
8 Graph CWEs with more than 10 CWE children

```
[98]: # ####################
      # Graph CWEs with more than 10 CWE children
      # ####################
      fig = plt.figure(figsize=(15,5))
      ax = sns.barplot(x="parent_name", y="child_count", palette="Blues_d",␣
       ↪data=cwe_child_count_gt_n, ci=None);
      plt.title('CWE Child count')
      plt.setp(ax.get_xticklabels(), rotation=90);
      plt.rc('xtick', labelsize=20)
      plt.rc('figure', titlesize=15)
```



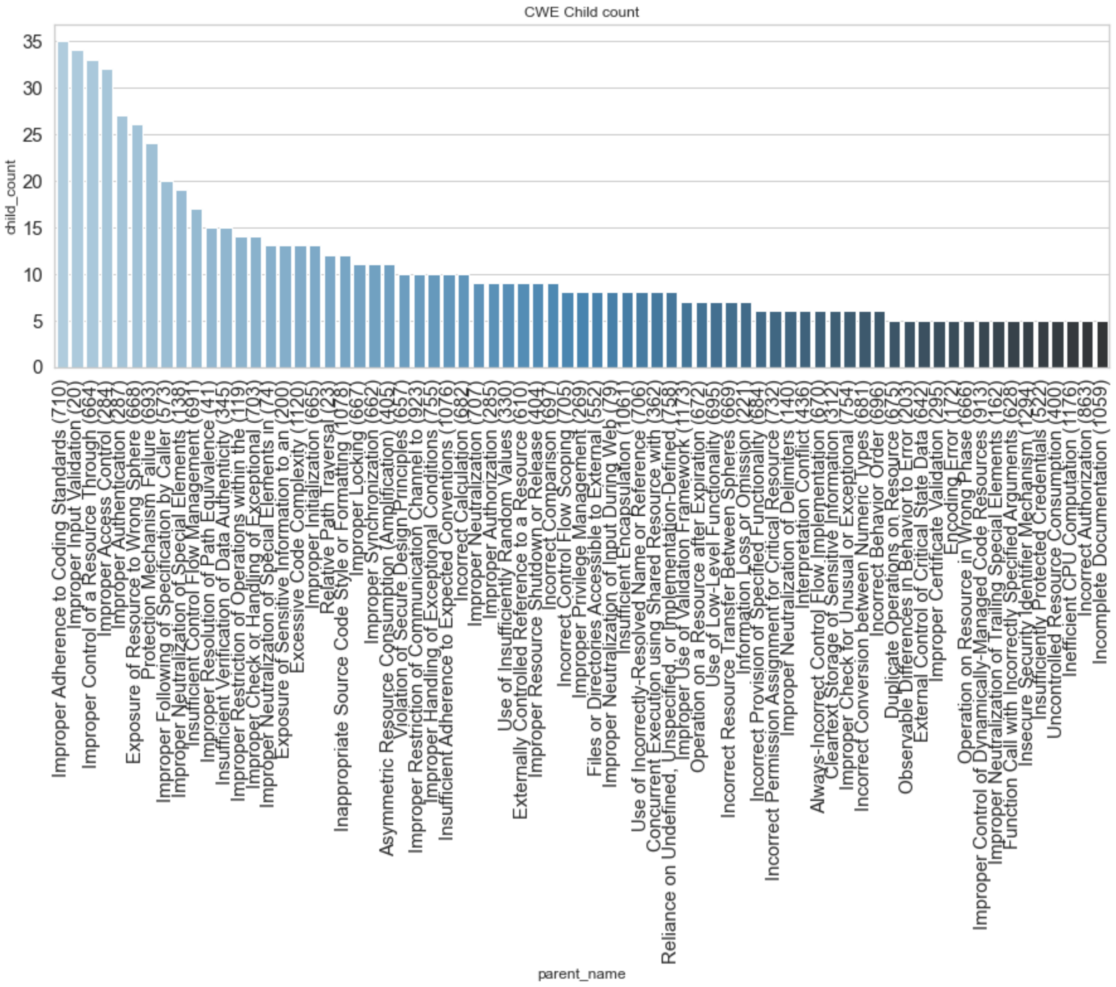
[99]:

```
fig = plt.figure(figsize=(15,5))
ax = sns.barplot(x="parent_name", y="child_count", palette="Blues_d",␣
 ↪data=cwe_child_count_gt_n2, ci=None);
plt.title('CWE Child count')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
```



[100]:

```
fig = plt.figure(figsize=(15,5))
ax = sns.barplot(x="parent_name", y="child_count", palette="Blues_d",␣
 ↪data=cwe_child_count_gt_n3, ci=None);
plt.title('CWE Child count')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
```



9 Create CWE Consequence Nodes

[101]:

```
# ####################
# Create CWE Consequence Nodes
# ####################
def combine_lists(list_1, list_2):
    final_result = set()
    for val1 in list_1:
        for val2 in list_2:
            final_result.add(f'{val1}-{val2}')
    return list(final_result)


consquences_set = set()
cwe_consequence_list = {}
```

```
consquences_incomming_cwe_count = {}

for obj in cwe['Weakness_Catalog']['Weaknesses']['Weakness']:
    cwe_id = obj['@ID']

    try:
        consequences = obj['Common_Consequences']['Consequence']
    except KeyError:
#         print(f'{cwe_id} does not have consequence obj')
#         print('----')
        continue

    if not isinstance(consequences, list):
        consequences_list = [consequences]
    else:
        consequences_list = consequences

    for cons in consequences_list:

        scope = cons.get('Scope', 'Unknown Scope')
        if not isinstance(scope, list):
            scope_list = [scope]
        else:
            scope_list = scope

        impact = cons.get('Impact', 'Unknown Impact')
        if not isinstance(impact, list):
            impact_list = [impact]
        else:
            impact_list = impact

        scope_impact_list = combine_lists(scope_list, impact_list)

        likelihood = cons.get('Likelihood', 'Unknown Likelihood')
        if not isinstance(likelihood, list):
            likelihood_list = [likelihood]
        else:
            likelihood_list = likelihood

        scope_impact_likelihood_list =␣
 ↪combine_lists(scope_impact_list,likelihood_list)

        for val in scope_impact_likelihood_list:
            if val not in consquences_set:
                consquences_set.add(val)
                cql_create_cwe_consequence_node = f"""CREATE (:consequence {{␣
 ↪consequence_id: "{val}"}})"""
```

```
                run_query(cql_create_cwe_consequence_node)

            cwe_cons_list = cwe_consequence_list.get(cwe_id, [])
            if val not in cwe_cons_list:
                cql_create_relationship = f"""MATCH (cwe1:cwe), (consequence1:␣
↪consequence)
                                      WHERE cwe1.cwe_id = '{cwe_id}' AND␣
↪consequence1.consequence_id = '{val}'
                                      CREATE (cwe1)-[r:causes]->(consequence1)
                                      RETURN type(r)"""
                run_query(cql_create_relationship)
                cwe_cons_list.append(val)
                cwe_consequence_list[cwe_id] = cwe_cons_list
                count = consquences_incomming_cwe_count.get(val, 0)
                count += 1
                consquences_incomming_cwe_count[val] = count
```

[102]:

```
for key,value in consquences_incomming_cwe_count.items():
    cql_update_cwe_node = f"""MATCH (c:consequence {{ consequence_id:␣
↪"{key}"}}) SET c.child_count = {value}"""
    run_query(cql_update_cwe_node)
```

[103]:

```
data = {'consequence_abbreviation': [], 'consequence_id':[], 'child_count':[]}
for key, value in consquences_incomming_cwe_count.items():
    cons_name = key.split('-')
    cons_name_abbr = f'{cons_name[0][:20]}-{cons_name[1][:20]}'

    data['consequence_abbreviation'].append(cons_name_abbr)
    data['consequence_id'].append(key)
    data['child_count'].append(value)

cwe_consequence_child_count_df = pd.DataFrame(data)
cwe_consequence_child_count_df.sort_values(by=['child_count'], ascending=False,␣
↪inplace=True)
cwe_consequence_child_count_df.reset_index(drop=True, inplace=True)
cwe_consequence_child_count_df_gt_n =␣
↪cwe_consequence_child_count_df[cwe_consequence_child_count_df['child_count']␣
↪>= 20]
```

[104]:

```
cwe_consequence_child_count_df_gt_n
```

[104]:

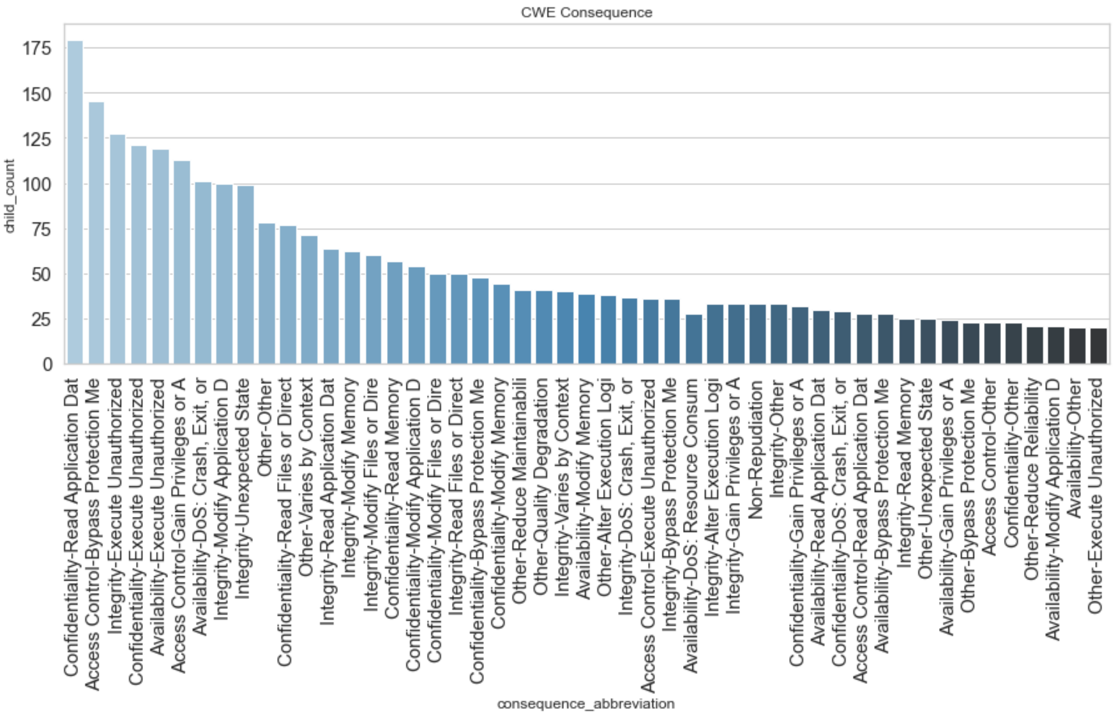
```
                consequence_abbreviation  \
0   Confidentiality-Read Application Dat
1    Access Control-Bypass Protection Me
2         Integrity-Execute Unauthorized
3   Confidentiality-Execute Unauthorized
4      Availability-Execute Unauthorized
```

```
5    Access Control-Gain Privileges or A
6      Availability-DoS: Crash, Exit, or
7         Integrity-Modify Application D
8             Integrity-Unexpected State
9                            Other-Other
10  Confidentiality-Read Files or Direct
11               Other-Varies by Context
12        Integrity-Read Application Dat
13               Integrity-Modify Memory
14        Integrity-Modify Files or Dire
15           Confidentiality-Read Memory
16  Confidentiality-Modify Application D
17  Confidentiality-Modify Files or Dire
18        Integrity-Read Files or Direct
19  Confidentiality-Bypass Protection Me
20         Confidentiality-Modify Memory
21            Other-Reduce Maintainabili
22             Other-Quality Degradation
23           Integrity-Varies by Context
24            Availability-Modify Memory
25            Other-Alter Execution Logi
26        Integrity-DoS: Crash, Exit, or
27   Access Control-Execute Unauthorized
28        Integrity-Bypass Protection Me
29     Availability-DoS: Resource Consum
30        Integrity-Alter Execution Logi
31        Integrity-Gain Privileges or A
32                       Non-Repudiation
33                       Integrity-Other
34  Confidentiality-Gain Privileges or A
35     Availability-Read Application Dat
36  Confidentiality-DoS: Crash, Exit, or
37   Access Control-Read Application Dat
38     Availability-Bypass Protection Me
39                 Integrity-Read Memory
40                Other-Unexpected State
41     Availability-DoS: Resource Consum
42     Availability-Gain Privileges or A
43            Other-Bypass Protection Me
44                  Access Control-Other
45     Availability-DoS: Resource Consum
46                 Confidentiality-Other
47              Other-Reduce Reliability
48     Availability-Modify Application D
49                    Availability-Other
50            Other-Execute Unauthorized
```

```
                                      consequence_id  child_count
0   Confidentiality-Read Application Data-Unknown …          179
1   Access Control-Bypass Protection Mechanism-Unk…          145
2   Integrity-Execute Unauthorized Code or Command…          127
3   Confidentiality-Execute Unauthorized Code or C…          121
4   Availability-Execute Unauthorized Code or Comm…          119
5   Access Control-Gain Privileges or Assume Ident…          113
6   Availability-DoS: Crash, Exit, or Restart-Unkn…          101
7   Integrity-Modify Application Data-Unknown Like…          100
8       Integrity-Unexpected State-Unknown Likelihood           99
9                      Other-Other-Unknown Likelihood           78
10  Confidentiality-Read Files or Directories-Unkn…           77
11         Other-Varies by Context-Unknown Likelihood           71
12  Integrity-Read Application Data-Unknown Likeli…           64
13         Integrity-Modify Memory-Unknown Likelihood           62
14  Integrity-Modify Files or Directories-Unknown …           60
15     Confidentiality-Read Memory-Unknown Likelihood           57
16  Confidentiality-Modify Application Data-Unknow…           54
17  Confidentiality-Modify Files or Directories-Un…           50
18  Integrity-Read Files or Directories-Unknown Li…           50
19  Confidentiality-Bypass Protection Mechanism-Un…           48
20   Confidentiality-Modify Memory-Unknown Likelihood           44
21    Other-Reduce Maintainability-Unknown Likelihood           41
22       Other-Quality Degradation-Unknown Likelihood           41
23     Integrity-Varies by Context-Unknown Likelihood           40
24      Availability-Modify Memory-Unknown Likelihood           39
25     Other-Alter Execution Logic-Unknown Likelihood           38
26  Integrity-DoS: Crash, Exit, or Restart-Unknown…           37
27  Access Control-Execute Unauthorized Code or Co…           36
28  Integrity-Bypass Protection Mechanism-Unknown …           36
29  Availability-DoS: Resource Consumption (CPU)-U…           36
30  Integrity-Alter Execution Logic-Unknown Likeli…           33
31  Integrity-Gain Privileges or Assume Identity-U…           33
32  Non-Repudiation-Hide Activities-Unknown Likeli…           33
33                 Integrity-Other-Unknown Likelihood           33
34  Confidentiality-Gain Privileges or Assume Iden…           32
35  Availability-Read Application Data-Unknown Lik…           30
36  Confidentiality-DoS: Crash, Exit, or Restart-U…           29
37  Access Control-Read Application Data-Unknown L…           28
38  Availability-Bypass Protection Mechanism-Unkno…           28
39           Integrity-Read Memory-Unknown Likelihood           25
40          Other-Unexpected State-Unknown Likelihood           25
41  Availability-DoS: Resource Consumption (Memory…           25
42  Availability-Gain Privileges or Assume Identit…           24
43  Other-Bypass Protection Mechanism-Unknown Like…           23
44            Access Control-Other-Unknown Likelihood           23
45  Availability-DoS: Resource Consumption (Other)…           23
```

```
46           Confidentiality-Other-Unknown Likelihood           23
47        Other-Reduce Reliability-Unknown Likelihood           21
48  Availability-Modify Application Data-Unknown L…           21
49              Availability-Other-Unknown Likelihood           20
50  Other-Execute Unauthorized Code or Commands-Un…           20
```

[105]:
```
fig = plt.figure(figsize=(15,5))
ax = sns.barplot(x="consequence_abbreviation", y="child_count",␣
 ↪palette="Blues_d", data=cwe_consequence_child_count_df_gt_n, ci=None);
plt.title('CWE Consequence')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
```



10 Create CWE Mode of Introduction Nodes

[106]:
```
# ####################
# Create CWE Mode of Introduction Nodes
# ####################
modes_of_intro_set = set()
modes_of_intro_cwe_list = {}
modes_of_intro_cwe_count = {}
```

```
for obj in cwe['Weakness_Catalog']['Weaknesses']['Weakness']:
    cwe_id = obj['@ID']
    try:
        modes_of_intro = obj['Modes_Of_Introduction']['Introduction']
    except KeyError:
#          print(f'{cwe_id} does not have modes_of_introduction')
        continue

    if not isinstance(modes_of_intro, list):
        modes_of_intro_list = [modes_of_intro]
    else:
        modes_of_intro_list = modes_of_intro

    for val in modes_of_intro_list:
        try:
            phase = val['Phase']
        except KeyError:
            continue
        except:
            print(val)
            raise ValueError

        if phase not in modes_of_intro_set:
            cql_create_cwe_modes_of_intro_node = f"""CREATE (:
↪mode_of_introduction {{ mode_of_intro_id: "{phase}"}})"""
            run_query(cql_create_cwe_modes_of_intro_node)
            modes_of_intro_set.add(phase)


        modes_of_intro_cwe_val = modes_of_intro_cwe_list.get(cwe_id, [])
        if phase not in modes_of_intro_cwe_val:
            cql_create_relationship = f"""MATCH (cwe1:cwe), (intro1:
↪mode_of_introduction)
                          WHERE cwe1.cwe_id = '{cwe_id}' AND intro1.
↪mode_of_intro_id = '{phase}'
                          CREATE (cwe1)-[r:introduced_in]->(intro1)
                          RETURN type(r)"""

            run_query(cql_create_relationship)
            count = modes_of_intro_cwe_count.get(phase, 0)
            count += 1
            modes_of_intro_cwe_count[phase] = count
            modes_of_intro_cwe_val.append(phase)
            modes_of_intro_cwe_list[cwe_id] = modes_of_intro_cwe_val
```

[107]:

```
for key,value in modes_of_intro_cwe_count.items():
    cql_update_cwe_node = f"""MATCH (c:mode_of_introduction {{ mode_of_intro_id:
 ↪ "{key}"}}) SET c.child_count = {value}"""
    run_query(cql_update_cwe_node)
```

[108]:

```
data = {'mode_of_introduction':[], 'child_count':[]}
for key, value in modes_of_intro_cwe_count.items():
    data['mode_of_introduction'].append(f'{key} ({value})')
    data['child_count'].append(value)

modes_of_intro_cwe_count_df = pd.DataFrame(data)
modes_of_intro_cwe_count_df.sort_values(by=['child_count'], ascending=False,␣
 ↪inplace=True)
modes_of_intro_cwe_count_df.reset_index(drop=True, inplace=True)
```

[109]:

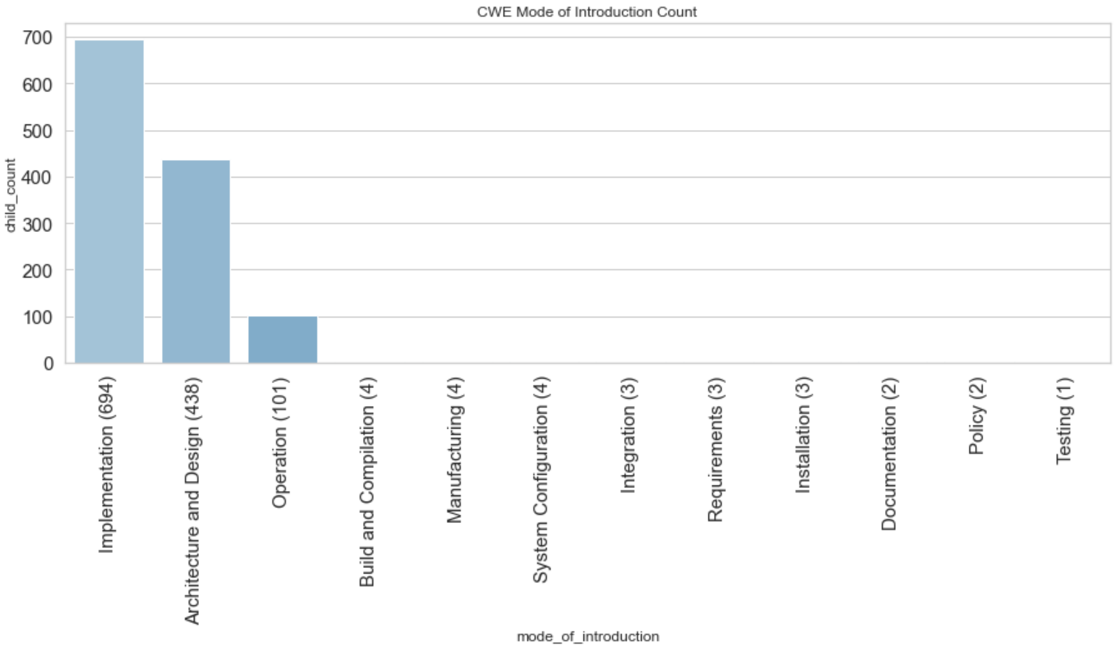
```
modes_of_intro_cwe_count_df
```

[109]:

```
             mode_of_introduction  child_count
0             Implementation (694)          694
1   Architecture and Design (438)          438
2                  Operation (101)          101
3        Build and Compilation (4)            4
4                Manufacturing (4)            4
5         System Configuration (4)            4
6                  Integration (3)            3
7                 Requirements (3)            3
8                 Installation (3)            3
9                Documentation (2)            2
10                      Policy (2)            2
11                     Testing (1)            1
```

[110]:

```
fig = plt.figure(figsize=(15,5))
ax = sns.barplot(x="mode_of_introduction", y="child_count", palette="Blues_d",␣
 ↪data=modes_of_intro_cwe_count_df, ci=None);
plt.title('CWE Mode of Introduction Count')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
```



11 CWE Time series

[111]:

```
# ####################
# CWE Time series
# ####################
data = {'cwe_id':[], 'cwe_name':[], 'submission_time':[]}

for obj in cwe['Weakness_Catalog']['Weaknesses']['Weakness']:
    cwe_id = obj['@ID']
    cwe_name = obj['@Name']
    submission_time = obj['Content_History']['Submission']['Submission_Date']
    submission_year = submission_time.split('-')[0]
    data['cwe_id'].append(cwe_id)
    data['cwe_name'].append(cwe_name)
    data['submission_time'].append(submission_year)

cwe_submission_time_df = pd.DataFrame(data)
```

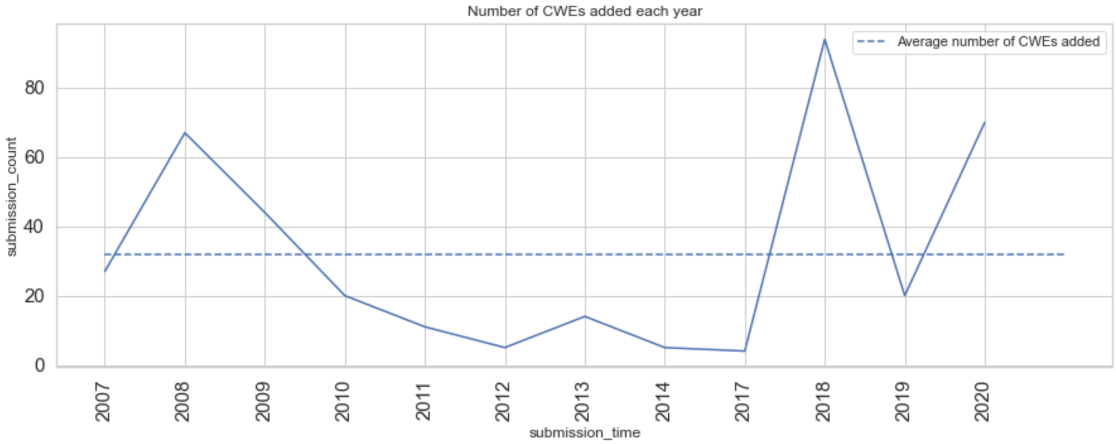
[112]:

```
cwe_submission_time_groupby = cwe_submission_time_df.
 ↪groupby(by=['submission_time']).count().reset_index(drop=False)
cwe_submission_time_groupby.rename({'cwe_id':
 ↪'submission_count'},axis='columns', inplace=True)
filtered_df =␣
 ↪cwe_submission_time_groupby[cwe_submission_time_groupby['submission_count']␣
 ↪< 200]
```

[113]:
```
filtered_df['submission_count'].describe()
```

[113]:
```
count    12.000000
mean     31.750000
std      30.115309
min       4.000000
25%       9.500000
50%      20.000000
75%      49.750000
max      94.000000
Name: submission_count, dtype: float64
```

[114]:
```
fig = plt.figure(figsize=(15,5))
ax = sns.lineplot(data=filtered_df, x="submission_time", y="submission_count");
plt.hlines(y=32, xmin=0, xmax=12, colors='b', linestyles='--', label='Average␣
 ↪number of CWEs added')
plt.title('Number of CWEs added each year')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
ax.legend();
```



[115]:
```
cwe_submission_time_groupby
```

[115]:
```
   submission_time  submission_count  cwe_name
0             2006               533       533
1             2007                27        27
2             2008                67        67
3             2009                44        44
4             2010                20        20
```

5	2011	11	11
6	2012	5	5
7	2013	14	14
8	2014	5	5
9	2017	4	4
10	2018	94	94
11	2019	20	20
12	2020	70	70

CVE_Analysis_2

December 6, 2020

[362]:

```python
import os
import json
from neo4j import GraphDatabase
import codecs
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set_theme(style="whitegrid")
import glob
from matplotlib.colors import ListedColormap
import numpy as np
```

[363]:

```python
base_dir = '/Users/janamian/Documents/workstation/ucsd_dse_program/fall_2019/
 ↪docker_vol/saba-ja/workstation/dse_203_2020/project/
 ↪dse_203_final_project_fall_2020/data'
```

[364]:

```python
nvdcve_files = sorted(glob.glob(os.path.join(base_dir, 'nvd_data','nvdcve-1.1*.
 ↪json')), reverse=True)
```

[365]:

```python
for val in nvdcve_files:
    print(val.split('/')[-1])
```

```
nvdcve-1.1-2020.json
nvdcve-1.1-2019.json
nvdcve-1.1-2018.json
nvdcve-1.1-2017.json
nvdcve-1.1-2016.json
nvdcve-1.1-2015.json
nvdcve-1.1-2014.json
nvdcve-1.1-2013.json
nvdcve-1.1-2012.json
nvdcve-1.1-2011.json
nvdcve-1.1-2010.json
nvdcve-1.1-2009.json
nvdcve-1.1-2008.json
nvdcve-1.1-2007.json
nvdcve-1.1-2006.json
nvdcve-1.1-2005.json
```

```
nvdcve-1.1-2004.json
nvdcve-1.1-2003.json
nvdcve-1.1-2002.json
```

[366]:

```
# #############
# Read all CWE data
# Read all NVD CVE Json files
# #############
with open(os.path.join(base_dir, 'cwe_data', 'cwec_v4.2.json')) as f:
    cwe = json.load(f)

nvd_list = []
for file_addr in nvdcve_files:
    with open(file_addr) as f:
        nvd_list.append(json.load(f))
```

[367]:

```
def trendline(xd, yd, order=1, c='r', alpha=1, Rval=False):
    """Make a line of best fit"""

    #Calculate trendline
    coeffs = np.polyfit(xd, yd, order)

    intercept = coeffs[-1]
    slope = coeffs[-2]
    power = coeffs[0] if order == 2 else 0

    minxd = np.min(xd)
    maxxd = np.max(xd)

    xl = np.array([minxd, maxxd])
    yl = power * xl ** 2 + slope * xl + intercept

    #Plot trendline
    plt.plot(xl, yl, c, alpha=alpha, linestyle='--')

    #Calculate R Squared
    p = np.poly1d(coeffs)

    ybar = np.sum(yd) / len(yd)
    ssreg = np.sum((p(xd) - ybar) ** 2)
    sstot = np.sum((yd - ybar) ** 2)
    Rsqr = ssreg / sstot

    if not Rval:
        #Plot R^2 value
        plt.text(0.8 * maxxd + 0.2 * minxd, 0.65 * np.max(yd) + 0.4 * np.
 ↪min(yd),
                 f'R^2 = {Rsqr:0.2f}\nm = {slope:0.0f}')
    else:
        #Return the R^2 value:
        return Rsqr
```

[368]:

```
def get_related_cwe(data_list):
    # CVE object
    resultw = []
    if not isinstance(data_list['problemtype']['problemtype_data'], list):
        print(data_list['problemtype']['problemtype_data'])
        raise ValueError

    if len(data_list['problemtype']['problemtype_data']) != 1:
        print(data_list['problemtype']['problemtype_data'])
        raise ValueError

    for val in data_list['problemtype']['problemtype_data'][0]['description']:
        resultw.append(val['value'])
    return resultw

def get_reference_url(data_list):
    result = []
    for val in data_list['references']['reference_data']:
        result.append(val['url'])
    return result

def get_tags(data_list):
    result = []
    for val in data_list['references']['reference_data']:
        for val2 in val['tags']:
            result.append(val2)

    return result

def get_description_data(data_list):
    result = []
    for val in data_list['description']['description_data']:
        if val['lang'] == 'en':
            result.append(val['value'])
    return result

def get_cpe_match(cpe_match_list):
    result = []
    try:
        for val in cpe_match_list['cpe_match']:
            result.append(val['cpe23Uri'])
    except KeyError:
        pass
    return result

def get_impacted_configuration(data_list):
    result = []
    for val in data_list['nodes']:

        result.extend(get_cpe_match(val))

        if 'children' in val.keys():
            for val2 in val['children']:
                result.extend(get_cpe_match(val2))

    return result

cve_clean_result = []
total_cwes = 0
total_cves = 0
for nvd_obj in nvd_list:
    for cve_obj in nvd_obj['CVE_Items']:
        published_date = cve_obj['publishedDate']
        yy = published_date.split('-')[0]
        if int(yy) < 2000:
            continue

        modified_date = cve_obj['lastModifiedDate']

        cve_id = cve_obj['cve']['CVE_data_meta']['ID']
        total_cves += 1

        related_cwe_list = get_related_cwe(cve_obj['cve'])
        if len(related_cwe_list) == 0:
            related_cwe_list = ['NVD-no-analysis']
            total_cwes += 1
#             print(cve_id)
        else:
            total_cwes += len(related_cwe_list)

        description = get_description_data(cve_obj['cve'])
        reference_url = get_reference_url(cve_obj['cve'])
        tags = get_tags(cve_obj['cve'])

        try:
            cvss_base_score =␣
 ↪cve_obj['impact']['baseMetricV3']['cvssV3']['baseScore']
            cvss_base_severity =␣
 ↪cve_obj['impact']['baseMetricV3']['cvssV3']['baseSeverity']
```

```
        except KeyError:
            cvss_base_score = 'unknown'
            cvss_base_severity = 'unknown'

        impacted_config = get_impacted_configuration(cve_obj['configurations'])

        cve_clean_result.append({
            'cve_id': cve_id,
            'related_cwe_list':related_cwe_list,
            'description': description,
            'reference_url':reference_url,
            'tags':tags,
            'cvss_base_score': cvss_base_score,
            'cvss_base_severity':cvss_base_severity,
            'impacted_config': impacted_config,
            'published_date': published_date,
            'modified_date': modified_date
         })
```

[369]:

```
counter = {}
for val in cve_clean_result:
    year = val['published_date'].split('-')[0]
    c = counter.get(year, 0)
    c += 1
    counter[year] = c

data = {'year':[], 'count':[]}
for key, value in counter.items():
    data['year'].append(int(key))
    data['count'].append(value)

cve_count_df = pd.DataFrame(data)
cve_count_df.sort_values(by=['year'], ascending=True, inplace=True)
cve_count_df.rename(columns={'count':'number_of_reported_cve'}, inplace=True)
cve_count_df.set_index('year', drop=True, inplace=True)
```

[370]:

```
cve_count_df
```

[370]:

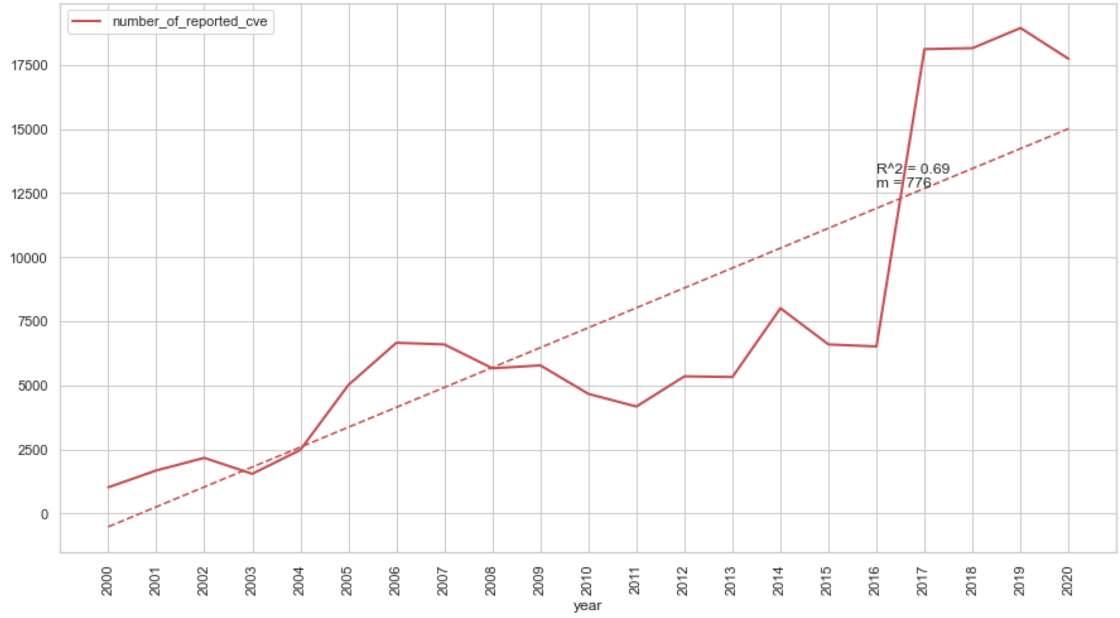
```
      number_of_reported_cve
year
2000                    1020
2001                    1679
2002                    2170
2003                    1548
2004                    2479
2005                    5010
2006                    6659
```

```
2007                     6596
2008                     5664
2009                     5778
2010                     4667
2011                     4172
2012                     5351
2013                     5324
2014                     8008
2015                     6595
2016                     6517
2017                    18113
2018                    18154
2019                    18938
2020                    17736
```

[371]:

```
print(cve_count_df['number_of_reported_cve'].sum())
cve_count_df.plot(kind='line', color='r',figsize=(15,8), linewidth=2)
plt.xticks(cve_count_df.index,rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
trendline(list(cve_count_df.index),␣
 ↪list(cve_count_df['number_of_reported_cve']), c='r')
```

```
152178
```



[372]:

```
cve_count_df.index
```

[372]:

```
Int64Index([2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010,
            2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020],
           dtype='int64', name='year')
```

[373]:

```
print('Total CVEs: ', len(cve_clean_result))
print('Total CWEs in CVEs: ', total_cwes)
```

```
Total CVEs:  152178
Total CWEs in CVEs:  154169
```

[374]:

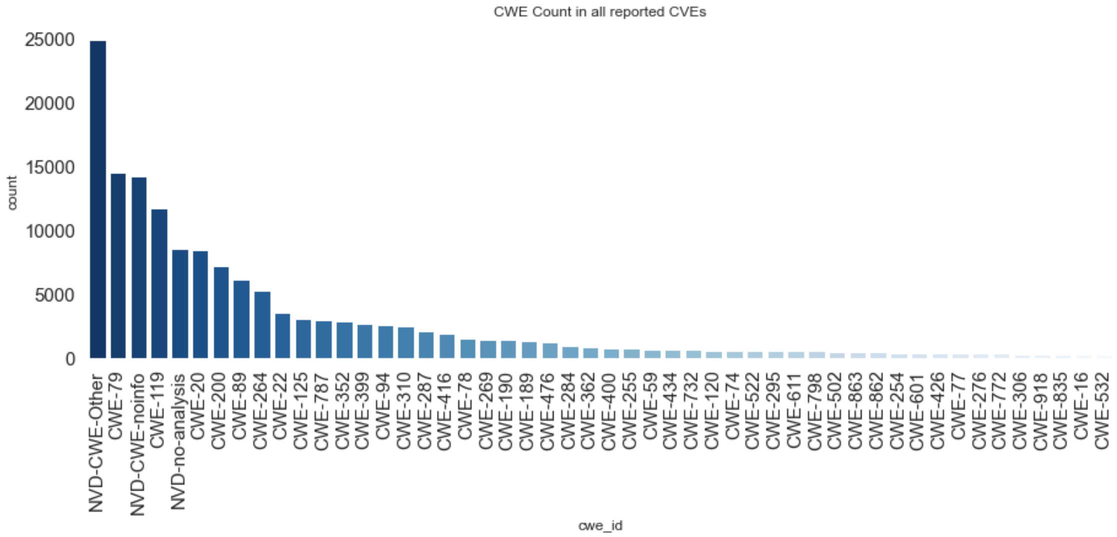
```
# #############
# Count CWEs causing CVE
# #############

cwe_count = {}
for val in cve_clean_result:
    for cwe in val['related_cwe_list']:
        cwe_c = cwe_count.get(cwe, 0)
        cwe_c += 1
        cwe_count[cwe] = cwe_c
```

[375]:

```
data = {'cwe_id':[], 'count':[]}
for key, value in cwe_count.items():
    data['cwe_id'].append(f'{key}')
    data['count'].append(value)

cwe_count_df = pd.DataFrame(data)
cwe_count_df.sort_values(by=['count'], ascending=False, inplace=True)
cwe_count_df.reset_index(drop=True, inplace=True)
cwe_count_filtered_gt_n = cwe_count_df[cwe_count_df['count'] >= 250]
```

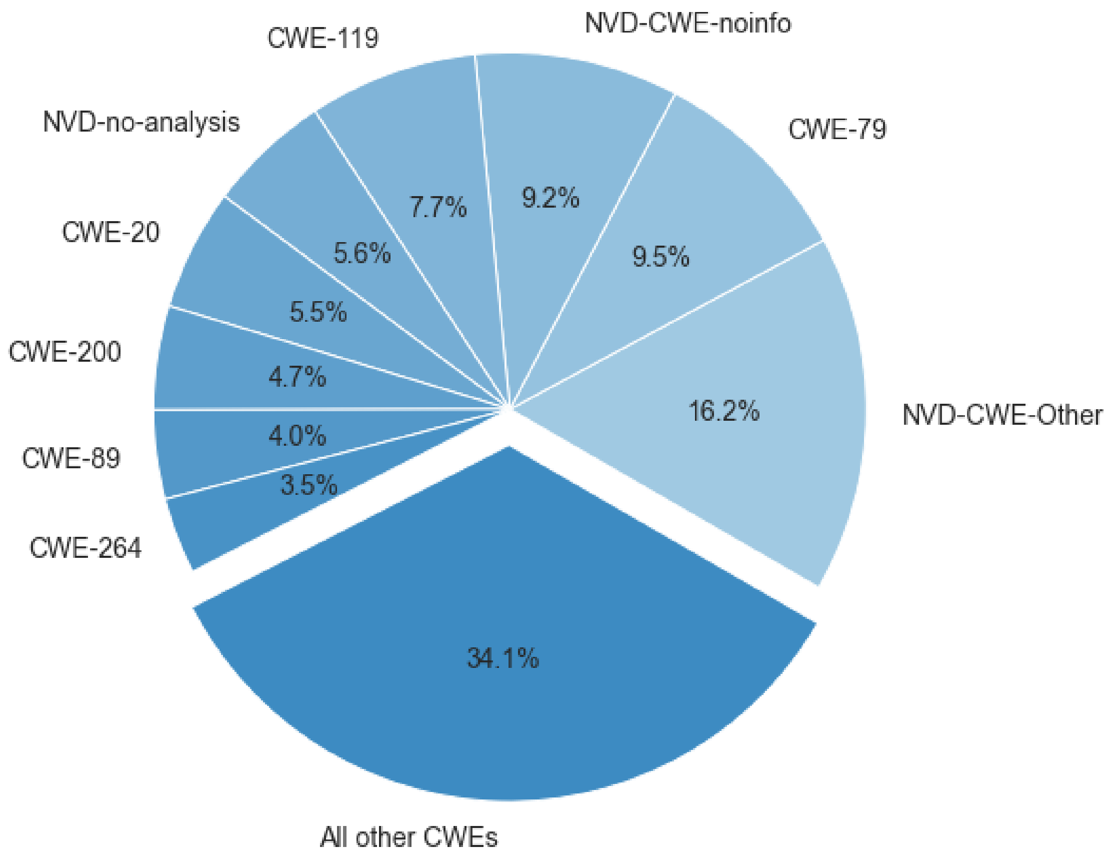
[376]:

```
fig = plt.figure(figsize=(15,5))
ax = sns.barplot(x="cwe_id", y="count", palette="Blues_r",␣
 ↪data=cwe_count_filtered_gt_n, ci=None);
plt.title('CWE Count in all reported CVEs')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
plt.grid(False)
plt.box(on=None)
```



[377]:
```
less_than_n_cwe = cwe_count_df[cwe_count_df['count'] < 5000]['count'].sum()
```

[378]:
```
labels = list(cwe_count_df[cwe_count_df['count'] >= 5000]['cwe_id'])
labels.append('All other CWEs')
sizes = list(cwe_count_df[cwe_count_df['count'] >= 5000]['count'])
sizes.append(less_than_n_cwe)
fig1, ax1 = plt.subplots(figsize=(8,8))
ax1.pie(sizes,
        explode=(0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1),
        labels=labels,
        autopct='%1.1f%%',
        shadow=False,
        startangle=-30,
        textprops={'fontsize': 14},
        colors=sns.color_palette("Blues_d", 20))

ax1.axis('equal')

plt.show()
```



[379]:
```
sum(sizes)
```

[379]: 154169

[380]:
```
column_names = []
column_names.extend(labels)
cve_to_cwe_per_year_df = pd.DataFrame(columns=column_names,␣
 ↪index=list(range(1988,2021)))
cve_to_cwe_per_year_df.fillna(0, inplace=True)
for val in cve_clean_result:
    year_val = int(val['published_date'].split('-')[0])
    for cwe in val['related_cwe_list']:
        if cwe in column_names:
            cwe_count = cve_to_cwe_per_year_df.loc[year_val, cwe]
            cve_to_cwe_per_year_df.loc[year_val, cwe] = cwe_count + 1
        else:
            cwe_count = cve_to_cwe_per_year_df.loc[year_val, 'All other CWEs']
            cve_to_cwe_per_year_df.loc[year_val, 'All other CWEs'] = cwe_count␣
 ↪+ 1
```

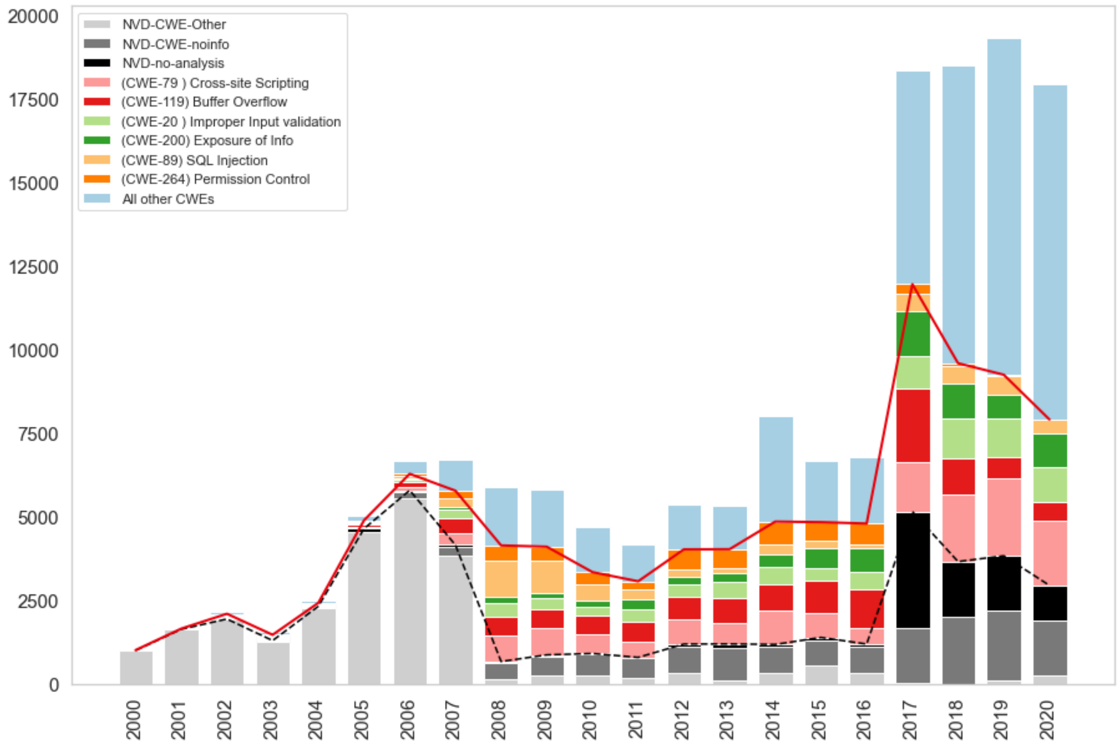
```
cve_to_cwe_per_year_df.rename(columns={
'CWE-79' : '(CWE-79 ) Cross-site Scripting',
'CWE-119': '(CWE-119) Buffer Overflow',
'CWE-20' : '(CWE-20 ) Improper Input validation',
'CWE-200': '(CWE-200) Exposure of Info',
'CWE-89' : '(CWE-89 ) SQL Injection',
'CWE-264': '(CWE-264) Permission Control'}, inplace=True)
```

[381]:
```
df_temp1 = cve_to_cwe_per_year_df[cve_to_cwe_per_year_df.index >=␣
 ↪2000][['NVD-CWE-Other', 'NVD-CWE-noinfo', 'NVD-no-analysis']]

line_data = df_temp1.sum(axis=1)
line_data_df_1 = line_data.to_frame().reset_index(drop=False)
line_data_df_1.rename(columns={'index':'year', 0:'NVD-CWE-Other'}, inplace=True)
line_data_df_1.astype(int)
line_data_df_1.set_index('year', drop=True, inplace=True)
NVD_CWE_Other_only = line_data_df_1
```

[382]:
```
df_temp2 = cve_to_cwe_per_year_df[cve_to_cwe_per_year_df.index >= 2000].
 ↪drop(['All other CWEs'], axis=1)
line_data = df_temp2.sum(axis=1)
line_data_df_2 = line_data.to_frame().reset_index(drop=False)
line_data_df_2.rename(columns={'index':'year', 0:'cwe_count'}, inplace=True)
line_data_df_2.astype(int)
line_data_df_2.set_index('year', drop=True, inplace=True)
majro_cwe_each_year = line_data_df_2
```

[383]:
```
width=0.75
filtered_years = cve_to_cwe_per_year_df[cve_to_cwe_per_year_df.index >= 2000]
fig, ax = plt.subplots(figsize=(15,10))
years = list(filtered_years.index)

nvd_1 = filtered_years['NVD-CWE-Other']
nvd_2 = filtered_years['NVD-CWE-noinfo']
nvd_3 = filtered_years['NVD-no-analysis']

cwe_79 = filtered_years['(CWE-79 ) Cross-site Scripting']
cwe_119 = filtered_years['(CWE-119) Buffer Overflow']
cwe_20 = filtered_years['(CWE-20 ) Improper Input validation']
cwe_200 = filtered_years['(CWE-200) Exposure of Info']
cwe_89 = filtered_years['(CWE-89 ) SQL Injection']
cwe_264 = filtered_years['(CWE-264) Permission Control']
all_other_cwe = filtered_years['All other CWEs']

ax.bar(years, nvd_1, width, label='NVD-CWE-Other', color='#CFCFCF') # '#B4D47B'
ax.bar(years, nvd_2, width, bottom=sum([nvd_1]), label='NVD-CWE-noinfo',␣
 ↪color='#797979')
```

```
ax.bar(years, nvd_3, width, bottom=sum([nvd_1, nvd_2]),␣
 ↪label='NVD-no-analysis', color='#000000')

ax.bar(years, cwe_79, width, bottom=sum([nvd_1, nvd_2, nvd_3]), label='(CWE-79␣
 ↪) Cross-site Scripting', color='#FB9A99')
ax.bar(years, cwe_119, width, bottom=sum([nvd_1, nvd_2, nvd_3, cwe_79]),␣
 ↪label='(CWE-119) Buffer Overflow', color='#E31B1B')
ax.bar(years, cwe_20, width, bottom=sum([nvd_1, nvd_2, nvd_3, cwe_79,␣
 ↪cwe_119]), label='(CWE-20 ) Improper Input validation', color='#B3DF89')
ax.bar(years, cwe_200, width, bottom=sum([nvd_1, nvd_2, nvd_3, cwe_79, cwe_119,␣
 ↪cwe_20]), label='(CWE-200) Exposure of Info', color='#33A02C')
ax.bar(years, cwe_89, width, bottom=sum([nvd_1, nvd_2, nvd_3, cwe_79, cwe_119,␣
 ↪cwe_20, cwe_200]), label='(CWE-89) SQL Injection', color='#FCC06F')
ax.bar(years, cwe_264, width, bottom=sum([nvd_1, nvd_2, nvd_3, cwe_79, cwe_119,␣
 ↪cwe_20, cwe_200, cwe_89]), label='(CWE-264) Permission Control',␣
 ↪color='#FF7F01')
ax.bar(years, all_other_cwe, width, bottom=sum([nvd_1, nvd_2, nvd_3, cwe_79,␣
 ↪cwe_119, cwe_20, cwe_200, cwe_89, cwe_264]), label='All other CWEs',␣
 ↪color='#A6CFE3')

ax.plot(years, NVD_CWE_Other_only, color='#000000', linestyle='--')
ax.plot(years, majro_cwe_each_year, color='#E8000B', linewidth=2)
ax.legend()
plt.xticks(years,rotation=90);
plt.grid(False)
# plt.box(on=None)
```



[384]:

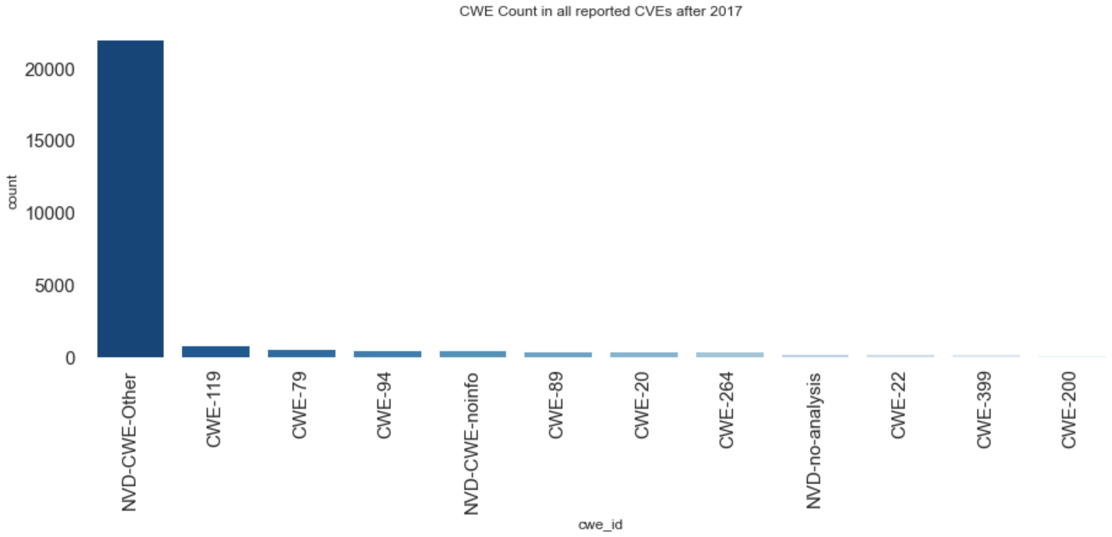
```
# #############
# Count CWEs causing CVE in 2017 and after only
# #############
def count_the_cwes(years):
    cwe_count = {}
    for val in cve_clean_result:
        year = val['published_date'].split('-')[0]
        if year not in years:
            continue
        for cwe in val['related_cwe_list']:
            cwe_c = cwe_count.get(cwe, 0)
            cwe_c += 1
            cwe_count[cwe] = cwe_c
    return cwe_count
```

[385]:

```
# #############
# CWEs color constants
# #############
NVDCWEOther_COLOR = '#CFCFCF'
NVDCWEnoinfo_COLOR = '#797979'
NVDnoanalysis_COLOR = '#000000'
```

[386]:
```
data = {'cwe_id':[], 'count':[]}
for key, value in count_the_cwes(['2000', '2001', '2002', '2003', '2004',␣
 ↪'2005', '2006', '2007']).items():
    data['cwe_id'].append(f'{key}')
    data['count'].append(value)

cwe_count_df = pd.DataFrame(data)
cwe_count_df.sort_values(by=['count'], ascending=False, inplace=True)
cwe_count_df.reset_index(drop=True, inplace=True)
cwe_count_filtered_gt_n = cwe_count_df[cwe_count_df['count'] >= 200]
```

[387]:
```
fig = plt.figure(figsize=(15,5))
ax = sns.barplot(x="cwe_id", y="count", palette="Blues_r",␣
 ↪data=cwe_count_filtered_gt_n, ci=None);
plt.title('CWE Count in all reported CVEs after 2017')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
plt.grid(False)
plt.box(on=None)
```



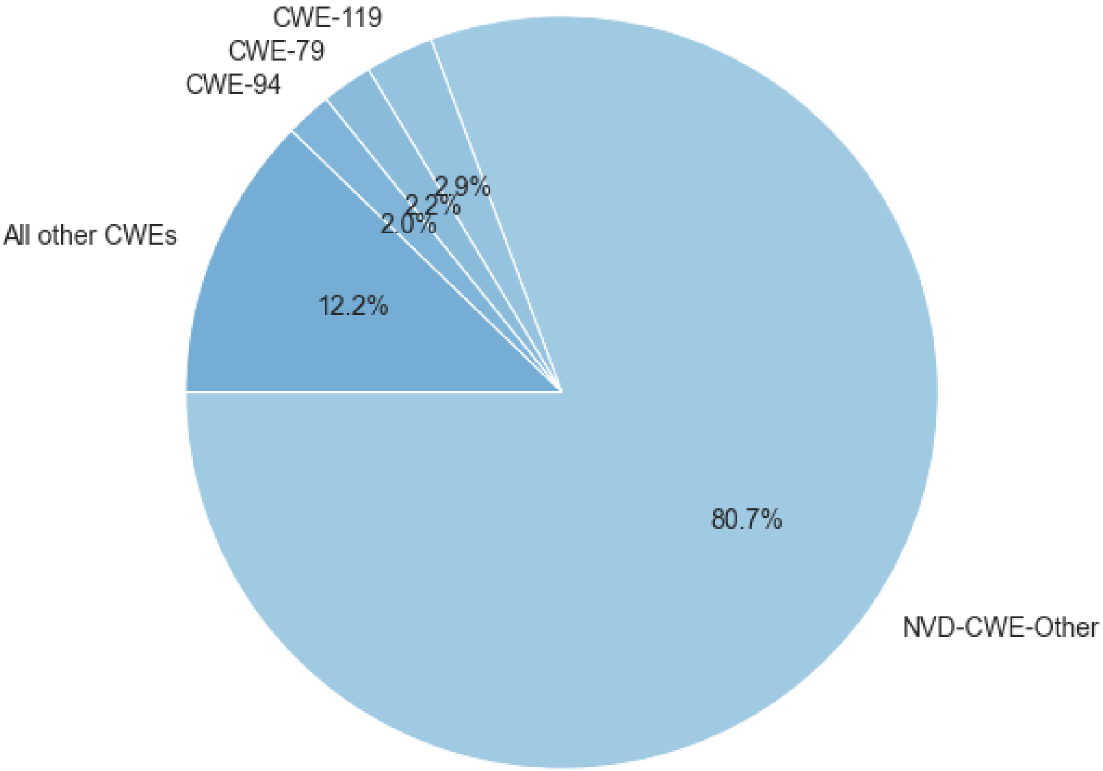
[388]:
```
len(sizes)
```

[388]:
```
10
```

[389]:
```
def graph_cwe_count_chart(limit, angle, colors=sns.color_palette("Blues_d",␣
 ↪20)):
    less_than_n_cwe = cwe_count_df[cwe_count_df['count'] < limit]['count'].sum()
    labels = list(cwe_count_df[cwe_count_df['count'] >= limit]['cwe_id'])
    labels.append('All other CWEs')
    sizes = list(cwe_count_df[cwe_count_df['count'] >= limit]['count'])
    sizes.append(less_than_n_cwe)
    fig1, ax1 = plt.subplots(figsize=(8,8))
    ax1.pie(sizes,
    #         explode=(0, 0, 0, 0, 0, 0, 0, 0, 0, 0.1),
            labels=labels,
            autopct='%1.1f%%',
            shadow=False,
            startangle=angle,
            textprops={'fontsize': 14},
            colors=colors)

    ax1.axis('equal')

    plt.show()
```

[390]:

```
graph_cwe_count_chart(500, -180)
```

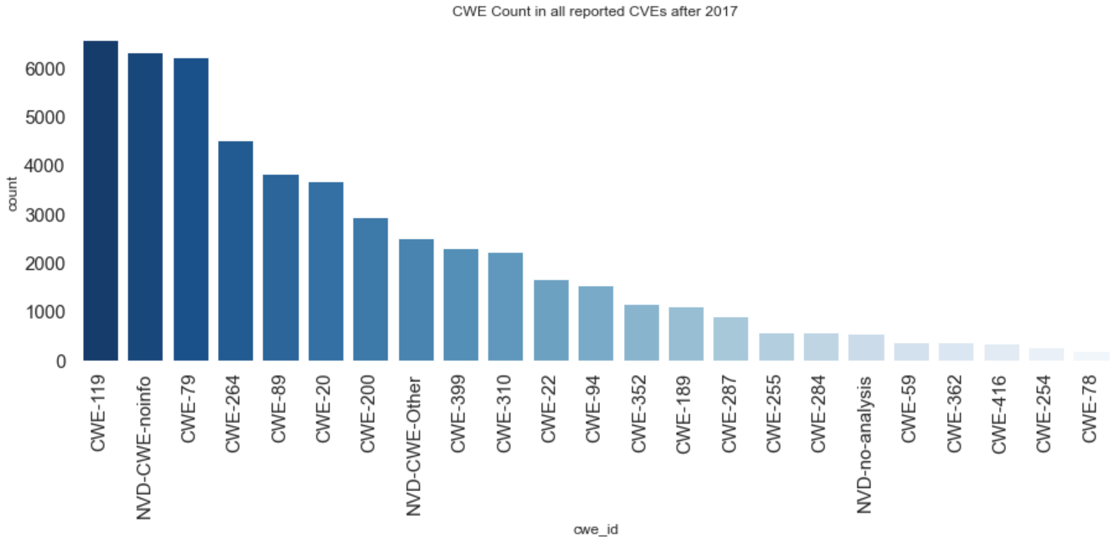


[391]:

```
data = {'cwe_id':[], 'count':[]}
for key, value in count_the_cwes(['2008', '2009', '2010', '2011',␣
 ↪'2012','2013', '2014', '2015', '2016']).items():
    data['cwe_id'].append(f'{key}')
    data['count'].append(value)

cwe_count_df = pd.DataFrame(data)
cwe_count_df.sort_values(by=['count'], ascending=False, inplace=True)
cwe_count_df.reset_index(drop=True, inplace=True)
cwe_count_filtered_gt_n = cwe_count_df[cwe_count_df['count'] >= 200]
```

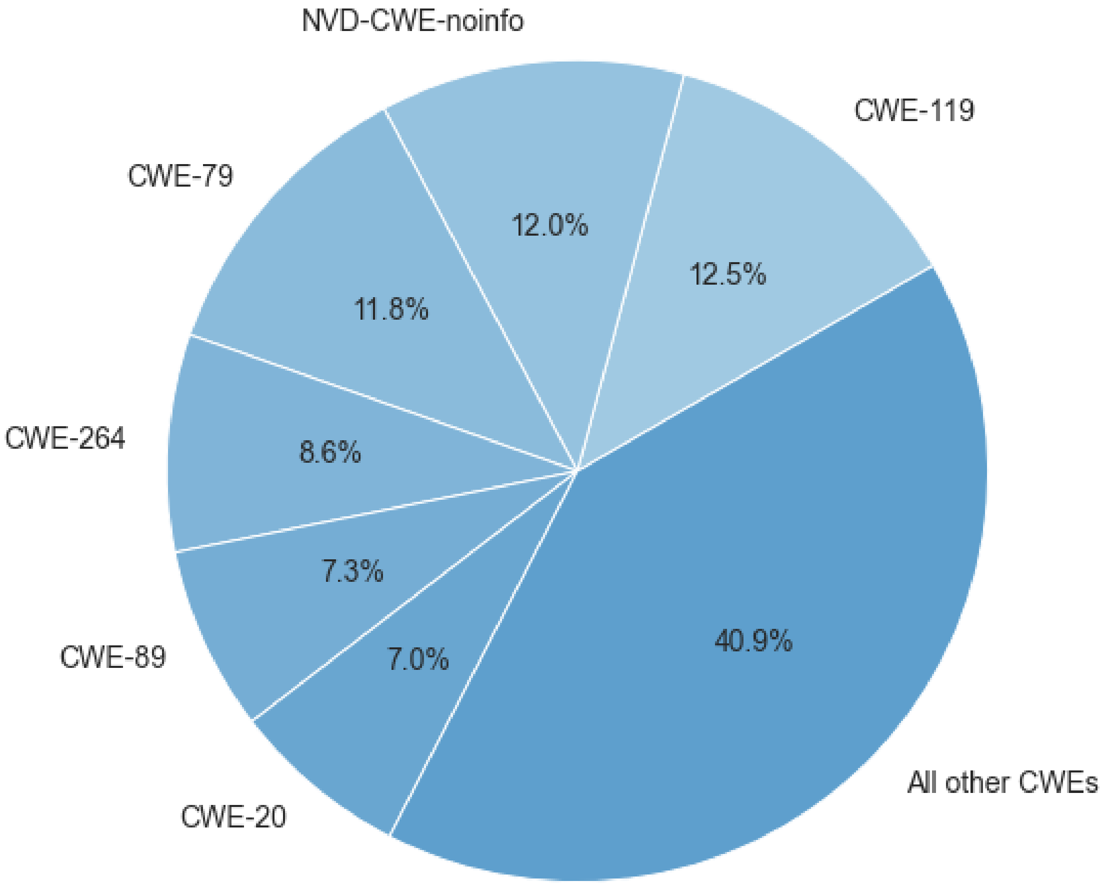
[392]:

```
fig = plt.figure(figsize=(15,5))
ax = sns.barplot(x="cwe_id", y="count", palette="Blues_r",␣
 ↪data=cwe_count_filtered_gt_n, ci=None);
plt.title('CWE Count in all reported CVEs after 2017')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
plt.grid(False)
plt.box(on=None)
```



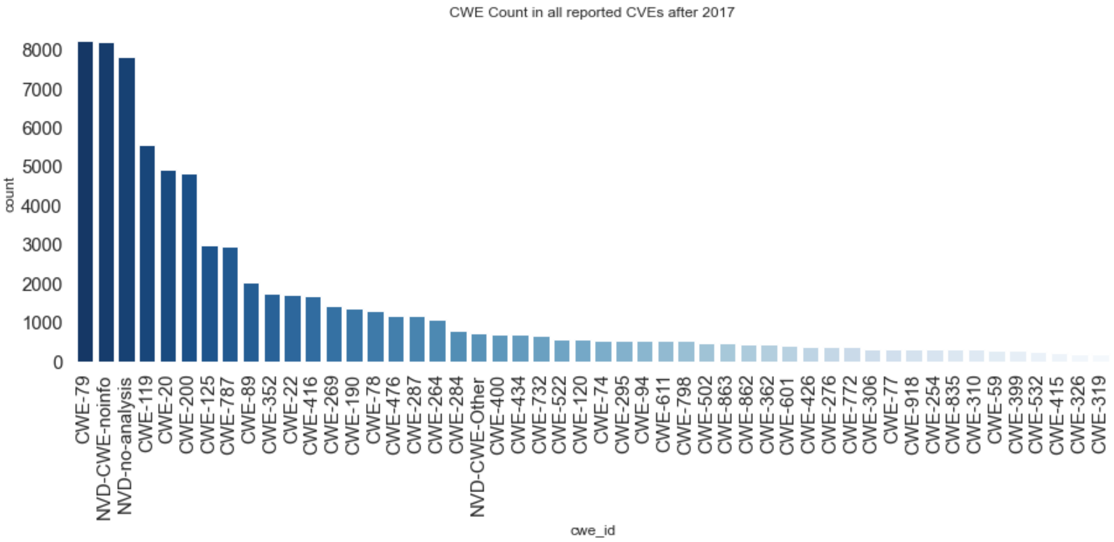
[393]:

```
graph_cwe_count_chart(3000, 30)
```

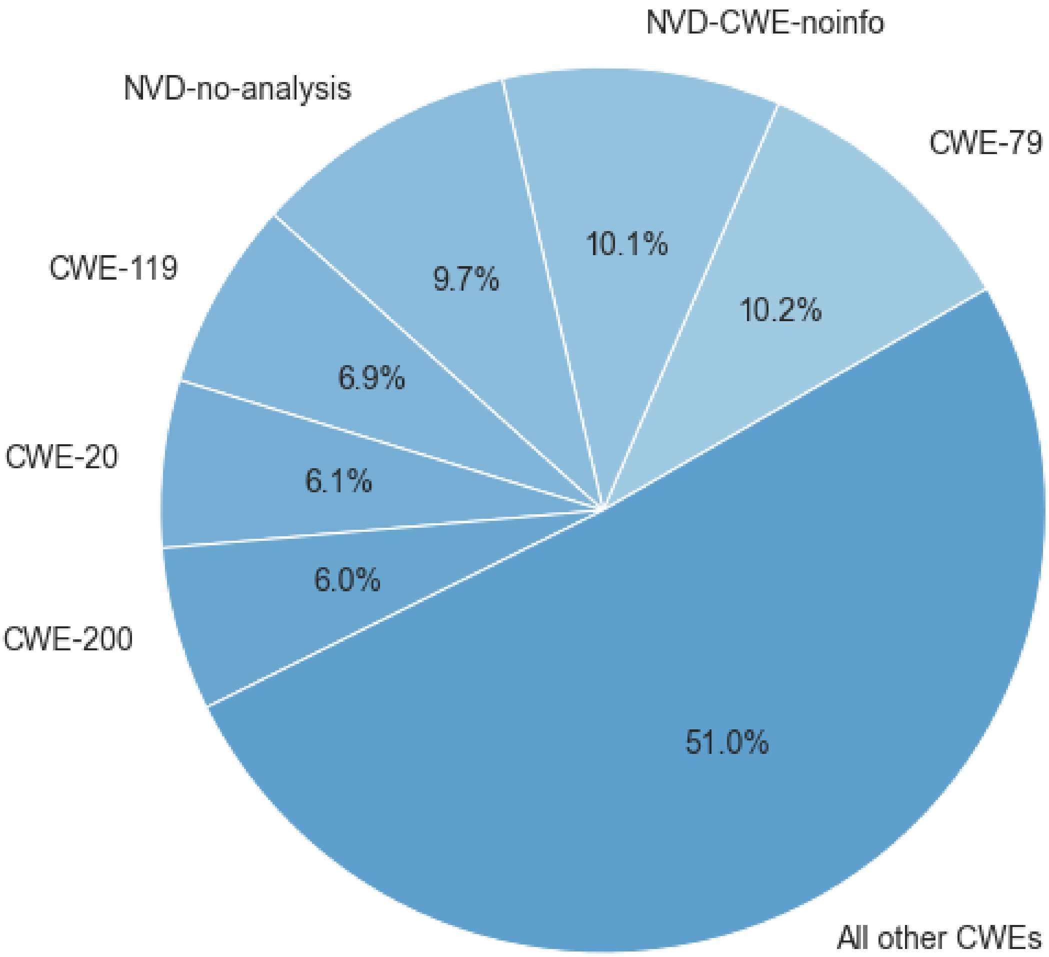


[394]:
```
data = {'cwe_id':[], 'count':[]}
for key, value in count_the_cwes(['2016', '2017', '2018', '2019','2020']).
 ↪items():
    data['cwe_id'].append(f'{key}')
    data['count'].append(value)

cwe_count_df = pd.DataFrame(data)
cwe_count_df.sort_values(by=['count'], ascending=False, inplace=True)
cwe_count_df.reset_index(drop=True, inplace=True)
cwe_count_filtered_gt_n = cwe_count_df[cwe_count_df['count'] >= 200]
```

[395]:
```
fig = plt.figure(figsize=(15,5))
ax = sns.barplot(x="cwe_id", y="count", palette="Blues_r",␣
 ↪data=cwe_count_filtered_gt_n, ci=None);
plt.title('CWE Count in all reported CVEs after 2017')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
plt.grid(False)
plt.box(on=None)
```



```
[396]: graph_cwe_count_chart(3000, 30)
```



[397]:

```python
# #############
# Companies reporting CVEs
# #############
company_counter = {}
product_counter = {}
company_impact_severity = {}
company_product_counter = {}

for val in cve_clean_result:
    impacted_cpe_list = val['impacted_config']
    impact_severity = val['cvss_base_severity']
    for val2 in impacted_cpe_list:
        company = val2.split(':')[3]
        product = val2.split(':')[4]
        company_product = f'{company}:{product}'
        c = company_counter.get(company, 0)
        c += 1
        company_counter[company] = c

        c = product_counter.get(product, 0)
        c += 1
        product_counter[product] = c

        c = company_product_counter.get(company_product, 0)
        c += 1
        company_product_counter[company_product] = c

        c_obj = company_impact_severity.get(company, {'LOW':0, 'MEDIUM':0,␣
 ↪'HIGH':0, 'CRITICAL':0, 'unknown':0})
        c_obj[impact_severity] = c_obj[impact_severity] + 1
        company_impact_severity[company] = c_obj
```

[398]:

```
data = {'company':[], 'count':[]}
for key, value in company_counter.items():
    data['company'].append(f'{key}')
    data['count'].append(value)

company_count_df = pd.DataFrame(data)
company_count_df.sort_values(by=['count'], ascending=False, inplace=True)
company_count_df.reset_index(drop=True, inplace=True)
company_count_filtered_gt_n = company_count_df[company_count_df['count'] >=␣
 ↪5000]
```

[399]:

```
print('Total impacted products: ', sum(company_count_df['count']))
```

```
Total impacted products:  1636445
```

[400]:

```
company_count_filtered_gt_n
```

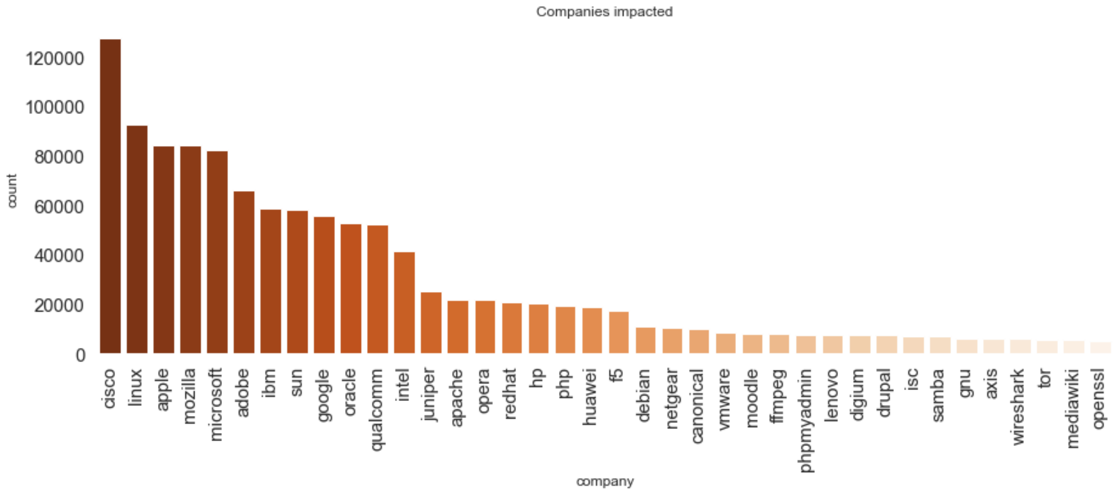
[400]:

```
      company   count
0       cisco  127417
1       linux   92648
2       apple   84128
3     mozilla   84066
4   microsoft   82450
5       adobe   66114
6         ibm   58708
7         sun   58173
8      google   55781
9      oracle   52521
10   qualcomm   52218
11      intel   41546
```

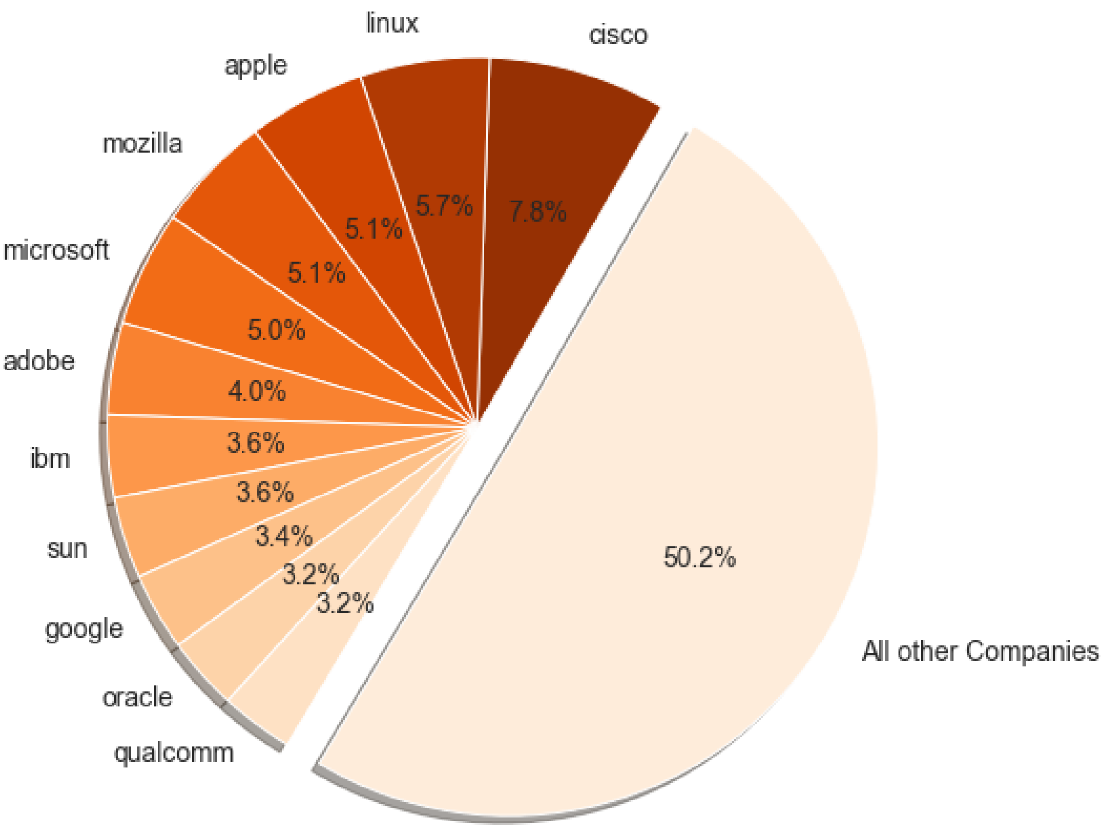
```
12     juniper  25022
13      apache  21794
14       opera  21704
15      redhat  20702
16          hp  20434
17         php  19284
18      huawei  18894
19          f5  17081
20      debian  10733
21     netgear  10460
22   canonical   9757
23      vmware   8563
24      moodle   7845
25      ffmpeg   7815
26  phpmyadmin   7641
27      lenovo   7529
28      digium   7442
29      drupal   7208
30         isc   6868
31       samba   6829
32         gnu   5908
33        axis   5808
34   wireshark   5754
35         tor   5411
36   mediawiki   5390
37     openssl   5013
```

[401]:

```
fig = plt.figure(figsize=(15,5))
ax = sns.barplot(x="company", y="count", palette="Oranges_r",␣
 ↪data=company_count_filtered_gt_n, ci=None);
# plt.yscale('log')
plt.title('Companies impacted')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
plt.grid(False)
plt.box(on=None)
```



```
[402]: less_than_n_cwe = company_count_df[company_count_df['count'] < 50000]['count'].
        ↪sum()
       labels = list(company_count_df[company_count_df['count'] >= 50000]['company'])
       labels.append('All other Companies')
       sizes = list(company_count_df[company_count_df['count'] >= 50000]['count'])
       sizes.append(less_than_n_cwe)
       explode = (0,0,0,0,0,0,0,0,0,0,0, 0.1)
       # pie_chart_color_list=["#53AAC0", "#53AACC", "#69C5E0", "#8DDBEB", "#D1F5FA"]
       # pie_chart_color_list=["#69C5EE", "#69C5E0", "#8DDBEB", "#D1F5FA","#69C5EE",␣
        ↪"#69C5E0", "#8DDBEB",  '#BFBFBD']
       # pie_chart_color_list=["#53AACC",'#ffcc99' ,'#66b3ff','#99ff99','#ff9999']

       fig1, ax1 = plt.subplots(figsize=(8,8))
       #
       # colors=pie_chart_color_list,

       ax1.pie(sizes, labels=labels, autopct='%1.1f%%', colors=sns.
        ↪color_palette("Oranges_r", 12),
               shadow=True, startangle=60,  textprops={'fontsize': 14},␣
        ↪explode=explode)
       ax1.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.

       plt.show()
       print(sum(sizes))
       print(labels)
```



```
1636445
['cisco', 'linux', 'apple', 'mozilla', 'microsoft', 'adobe', 'ibm', 'sun',
'google', 'oracle', 'qualcomm', 'All other Companies']
```

[403]:
```
data = {'product':[], 'count':[]}
for key, value in product_counter.items():
    data['product'].append(f'{key}')
    data['count'].append(value)

product_count_df = pd.DataFrame(data)
product_count_df.sort_values(by=['count'], ascending=False, inplace=True)
product_count_df.reset_index(drop=True, inplace=True)
product_count_filtered_gt_n = product_count_df[product_count_df['count'] >=␣
 ↪5000]
```

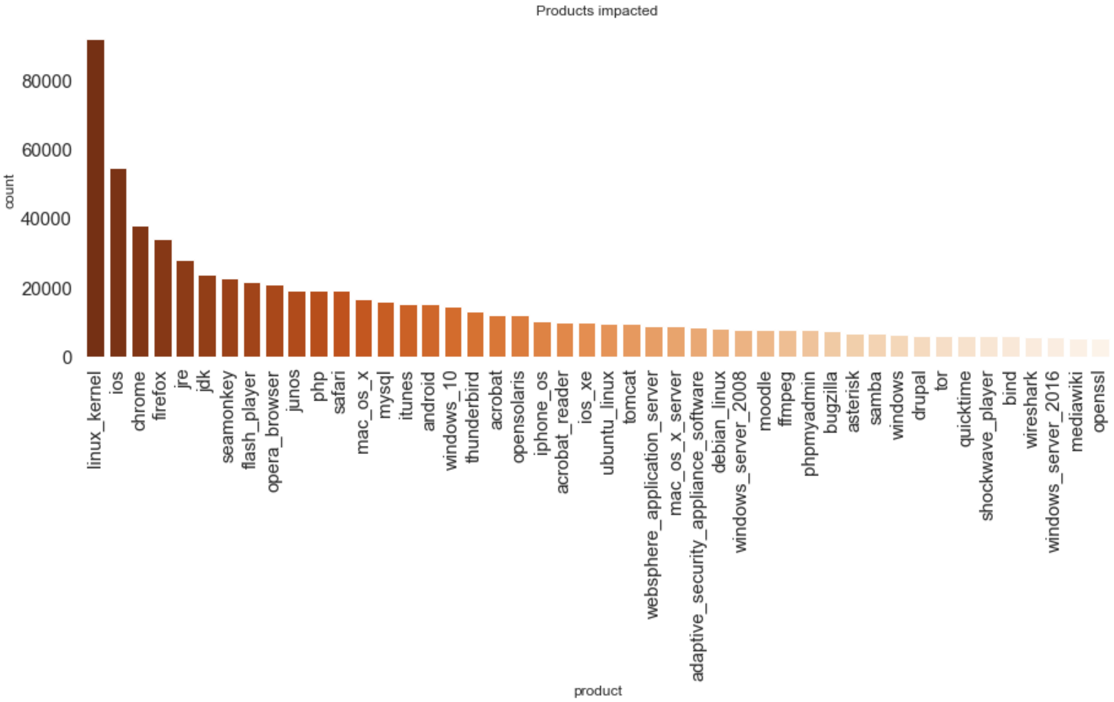
[404]:
```
product_count_filtered_gt_n
```

[404]:
```
                            product  count
0                      linux_kernel  92410
1                               ios  54799
```

```
2                                 chrome  38318
3                                firefox  34344
4                                    jre  28192
5                                    jdk  23900
6                              seamonkey  22808
7                           flash_player  21707
8                          opera_browser  20975
9                                  junos  19218
10                                   php  19180
11                                safari  19176
12                              mac_os_x  16568
13                                 mysql  16174
14                                itunes  15485
15                               android  15414
16                            windows_10  14651
17                           thunderbird  13259
18                               acrobat  12209
19                           opensolaris  11929
20                             iphone_os  10352
21                        acrobat_reader  10025
22                                ios_xe   9958
23                          ubuntu_linux   9604
24                                tomcat   9587
25           websphere_application_server   9059
26                        mac_os_x_server   8977
27  adaptive_security_appliance_software   8360
28                          debian_linux   8081
29                   windows_server_2008   7869
30                                moodle   7845
31                                ffmpeg   7810
32                            phpmyadmin   7644
33                              bugzilla   7316
34                              asterisk   6785
35                                 samba   6750
36                               windows   6289
37                                drupal   6188
38                                   tor   6174
39                             quicktime   6174
40                       shockwave_player   6124
41                                  bind   5878
42                             wireshark   5754
43                   windows_server_2016   5644
44                             mediawiki   5365
45                               openssl   5294
```

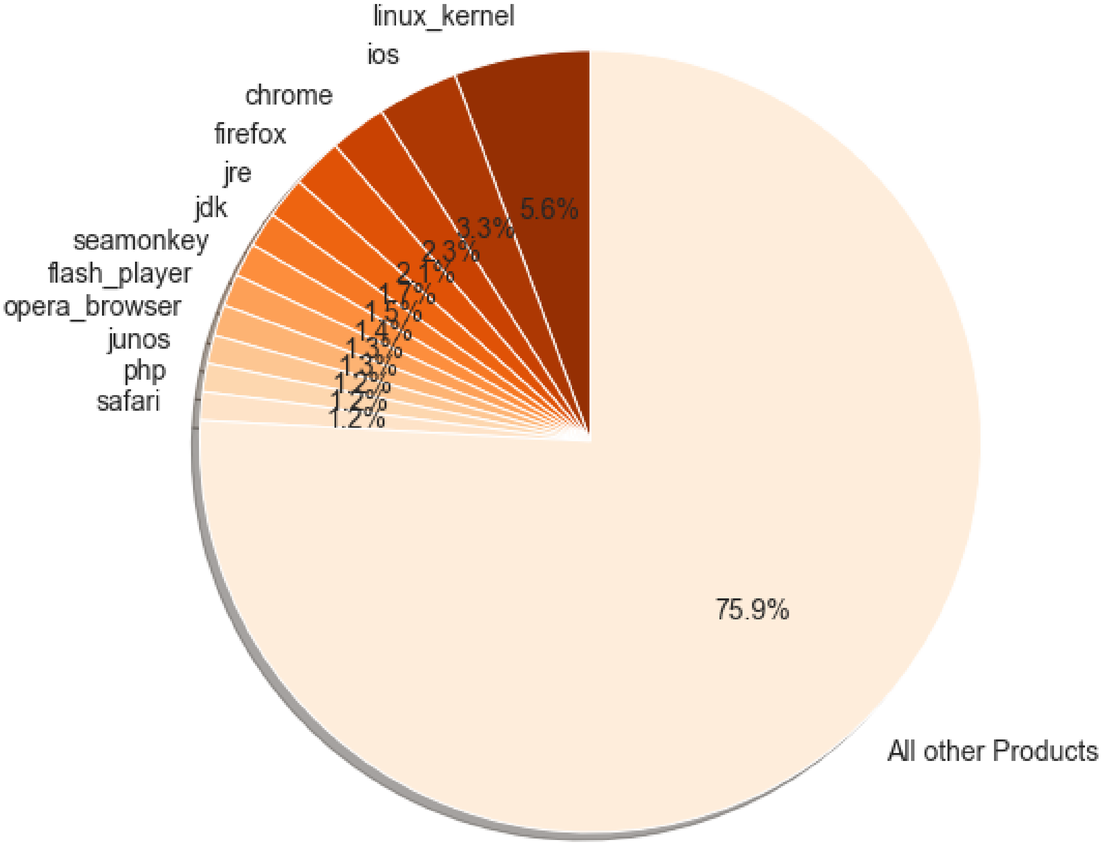
```
[405]: fig = plt.figure(figsize=(15,5))
```

```
ax = sns.barplot(x="product", y="count", palette="Oranges_r",␣
↪data=product_count_filtered_gt_n, ci=None);
# plt.yscale('log')
plt.title('Products impacted')
plt.setp(ax.get_xticklabels(), rotation=90);
plt.rc('xtick', labelsize=15)
plt.rc('ytick', labelsize=15)
plt.rc('figure', titlesize=15)
plt.grid(False)
plt.box(on=None)
```



[406]:
```
less_than_n_cwe = product_count_df[product_count_df['count'] < 17000]['count'].
↪sum()
labels = list(product_count_df[product_count_df['count'] >= 17000]['product'])
labels.append('All other Products')
sizes = list(product_count_df[product_count_df['count'] >= 17000]['count'])
sizes.append(less_than_n_cwe)
# explode = (0,0,0,0,0,0,0,0,0,0,0, 0.1)
# pie_chart_color_list=["#53AAC0", "#53AACC", "#69C5E0", "#8DDBEB", "#D1F5FA"]
# pie_chart_color_list=["#69C5EE", "#69C5E0", "#8DDBEB", "#D1F5FA","#69C5EE",␣
↪"#69C5E0", "#8DDBEB",  '#BFBFBD']
# pie_chart_color_list=["#53AACC",'#ffcc99' ,'#66b3ff','#99ff99','#ff9999']

fig1, ax1 = plt.subplots(figsize=(8,8))
```

```
#
# colors=pie_chart_color_list,

ax1.pie(sizes, labels=labels, autopct='%1.1f%%', colors=sns.
 ↪color_palette("Oranges_r",13),
        shadow=True, startangle=90,  textprops={'fontsize': 14}) #␣
 ↪explode=explode
ax1.axis('equal')

plt.show()
print(sum(sizes))
```



```
1636445
```

[514]:
```
most_impacted_comp = ['cisco', 'linux', 'apple', 'mozilla', 'microsoft',␣
 ↪'adobe', 'ibm', 'sun', 'google', 'oracle', 'qualcomm']

data = {'company': [],
        'CRITICAL': [],
        'HIGH': [],
        'MEDIUM': [],
        'LOW':[],
        'unknown': [],
        'Total_impacts_only': [],
        'Total_all': []}

for comp_name in company_impact_severity:
    if comp_name in most_impacted_comp:
        data['company'].append(comp_name)
        data['unknown'].append(company_impact_severity[comp_name]['unknown'])
        data['CRITICAL'].append(company_impact_severity[comp_name]['CRITICAL'])
        data['HIGH'].append(company_impact_severity[comp_name]['HIGH'])
        data['MEDIUM'].append(company_impact_severity[comp_name]['MEDIUM'])
        data['LOW'].append(company_impact_severity[comp_name]['LOW'])
        s1 = sum([company_impact_severity[comp_name]['LOW'],
                                 company_impact_severity[comp_name]['MEDIUM'],
                                 company_impact_severity[comp_name]['HIGH'],
                                ␣
 ↪company_impact_severity[comp_name]['CRITICAL']])
        s2 = company_impact_severity[comp_name]['unknown'] + s1
        data['Total_impacts_only'].append(s1)
        data['Total_all'].append(s2)
```

[515]:

```
company_impact_df = pd.DataFrame(data)
```

[517]:

```
company_impact_df.sort_values(by=['Total_impacts_only'], inplace=True,
                              ascending=False)
total_impacts_only = company_impact_df['Total_impacts_only']
total_all = company_impact_df['Total_all']

company_impact_df.drop(['Total_impacts_only', 'Total_all'], axis=1,␣
 ↪inplace=True)
```

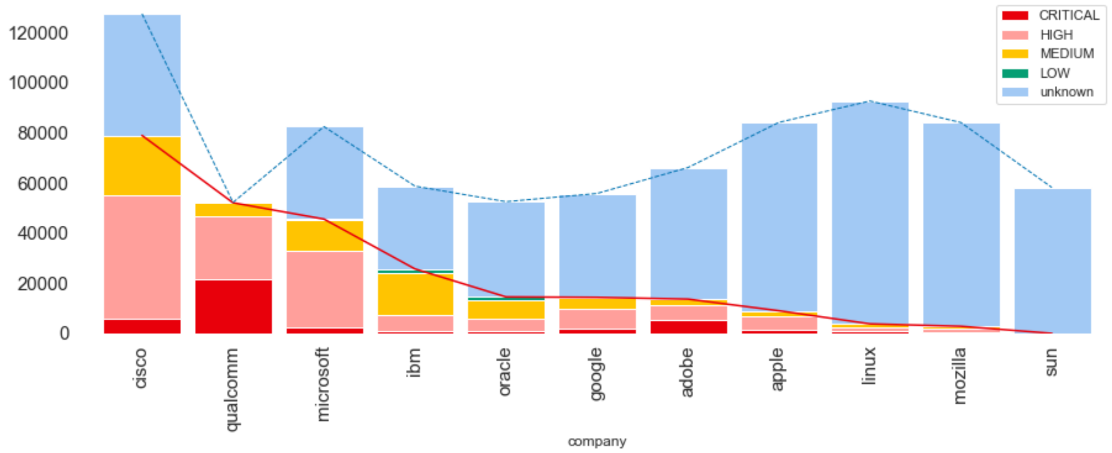
[518]:

```
company_impact_df.set_index('company', drop=True, inplace=True)
```

[535]:

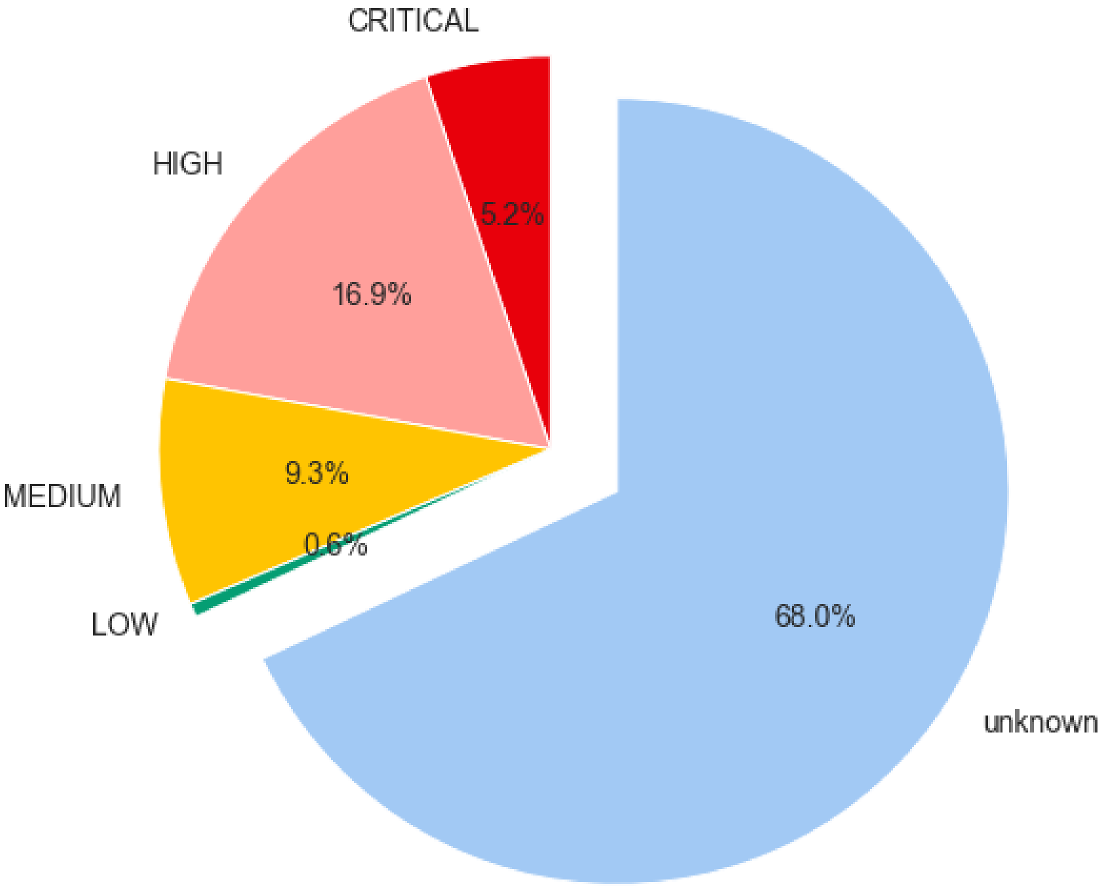
```
ax = company_impact_df.plot(
    kind='bar', stacked=True,
    colormap=ListedColormap(
        ['#E8000B',
         '#FF9F9B',
         '#FFC401',
         '#059E73',
        '#A2C9F4'
        ]), width=0.85, figsize=(15,5));
```

```
ax.plot(company_impact_df.index, total_impacts_only, color='#E8000B',␣
 ↪linestyle='-')
ax.plot(company_impact_df.index, total_all, color='#0073B2', linewidth=1,␣
 ↪linestyle='--')
plt.grid(False)
plt.box(on=None);
```



[540]:
```
sizes = [company_impact_df['CRITICAL'].sum(),
        company_impact_df['HIGH'].sum(),
        company_impact_df['MEDIUM'].sum(),
        company_impact_df['LOW'].sum(),
        company_impact_df['unknown'].sum()]
labels = ['CRITICAL', 'HIGH', 'MEDIUM', 'LOW', 'unknown']
fig1, ax1 = plt.subplots(figsize=(8,8))

ax1.pie(sizes, labels=labels, autopct='%1.1f%%', colors=['#E8000B',
         '#FF9F9B',
         '#FFC401',
         '#059E73',
        '#A2C9F4'
        ],
        explode=(0,0,0,0,0.2),
        shadow=False, startangle=90,  textprops={'fontsize': 14}) #␣
 ↪explode=explode
ax1.axis('equal');
print(sum(sizes))
```

```
814224
```



```
[528]: sizes
```

```
[528]: [42092, 137972, 75962, 4586, 553612]
```

CVE_Analysis_3

December 6, 2020

1 Creating CVE, CPE company, CPE SW, CVSS Nodes in Neo4j

[1]:
```
import os
import json
from neo4j import GraphDatabase
import codecs
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set_theme(style="whitegrid")
import glob
from matplotlib.colors import ListedColormap
import numpy as np
from tqdm.notebook import tqdm
```

2 Connect to Neo4j api

[2]:
```
uri = "neo4j://localhost:7687"
userName = "neo4j"
password = "password"
# Connect to the neo4j database server
graph_db_driver  = GraphDatabase.driver(uri, auth=(userName, password))
```

[3]:
```
def run_query(q):
    with graph_db_driver.session() as graph_db_session:
        try:
            graph_db_session.run(q)
        except:
            print(q)
            raise NameError
```

[4]:
```
base_dir = '/Users/janamian/Documents/workstation/ucsd_dse_program/fall_2019/
 ↪docker_vol/saba-ja/workstation/dse_203_2020/project/
 ↪dse_203_final_project_fall_2020/data'
```

[5]:

```
nvdcve_files = sorted(glob.glob(os.path.join(base_dir, 'nvd_data','nvdcve-1.1*.
 ↪json')), reverse=True)
```

3 Read NVD json files

[6]:

```
for val in nvdcve_files:
    print(val.split('/')[-1])
```

```
nvdcve-1.1-2020.json
nvdcve-1.1-2019.json
nvdcve-1.1-2018.json
nvdcve-1.1-2017.json
nvdcve-1.1-2016.json
nvdcve-1.1-2015.json
nvdcve-1.1-2014.json
nvdcve-1.1-2013.json
nvdcve-1.1-2012.json
nvdcve-1.1-2011.json
nvdcve-1.1-2010.json
nvdcve-1.1-2009.json
nvdcve-1.1-2008.json
nvdcve-1.1-2007.json
nvdcve-1.1-2006.json
nvdcve-1.1-2005.json
nvdcve-1.1-2004.json
nvdcve-1.1-2003.json
nvdcve-1.1-2002.json
```

[7]:

```
# #############
# Read all CWE data
# Read all NVD CVE Json files
# #############
with open(os.path.join(base_dir, 'cwe_data', 'cwec_v4.2.json')) as f:
    cwe = json.load(f)

nvd_list = []
for file_addr in nvdcve_files:
    with open(file_addr) as f:
        nvd_list.append(json.load(f))
```

4 Utility functions

[8]:

```python
def get_related_cwe(data_list):
    # CVE object
    resultw = []
    if not isinstance(data_list['problemtype']['problemtype_data'], list):
        print(data_list['problemtype']['problemtype_data'])
        raise ValueError

    if len(data_list['problemtype']['problemtype_data']) != 1:
        print(data_list['problemtype']['problemtype_data'])
        raise ValueError

    for val in data_list['problemtype']['problemtype_data'][0]['description']:
        resultw.append(val['value'])
    return resultw

def get_reference_url(data_list):
    result = []
    for val in data_list['references']['reference_data']:
        result.append(val['url'])
    return result

def get_tags(data_list):
    result = []
    for val in data_list['references']['reference_data']:
        for val2 in val['tags']:
            result.append(val2)

    return result

def get_description_data(data_list):
    result = []
    for val in data_list['description']['description_data']:
        if val['lang'] == 'en':
            result.append(val['value'])
    return result

def get_cpe_match(cpe_match_list):
    result = []
    try:
        for val in cpe_match_list['cpe_match']:
            result.append(val['cpe23Uri'])
    except KeyError:
        pass
    return result
```

```
def get_impacted_configuration(data_list):
    result = []
    for val in data_list['nodes']:

        result.extend(get_cpe_match(val))

        if 'children' in val.keys():
            for val2 in val['children']:
                result.extend(get_cpe_match(val2))

    return result

cve_clean_result = []
total_cwes = 0
total_cves = 0
for nvd_obj in nvd_list:
    for cve_obj in nvd_obj['CVE_Items']:
        published_date = cve_obj['publishedDate']
        yy = published_date.split('-')[0]
        if int(yy) < 2000:
            continue

        modified_date = cve_obj['lastModifiedDate']

        cve_id = cve_obj['cve']['CVE_data_meta']['ID']
        total_cves += 1

        related_cwe_list = get_related_cwe(cve_obj['cve'])
        if len(related_cwe_list) == 0:
            related_cwe_list = ['NVD-no-analysis']
            total_cwes += 1
#             print(cve_id)
        else:
            total_cwes += len(related_cwe_list)

        description = get_description_data(cve_obj['cve'])
        reference_url = get_reference_url(cve_obj['cve'])
        tags = get_tags(cve_obj['cve'])

        try:
            cvss_base_score =␣
 ↪cve_obj['impact']['baseMetricV3']['cvssV3']['baseScore']
            cvss_base_severity =␣
 ↪cve_obj['impact']['baseMetricV3']['cvssV3']['baseSeverity']
        except KeyError:
            cvss_base_score = -1
            cvss_base_severity = 'unknown'
        impacted_config = get_impacted_configuration(cve_obj['configurations'])

        cve_clean_result.append({
            'cve_id': cve_id,
            'related_cwe_list':related_cwe_list,
            'description': description,
            'reference_url':reference_url,
            'tags':tags,
            'cvss_base_score': cvss_base_score,
            'cvss_base_severity':cvss_base_severity,
            'impacted_config': impacted_config,
            'published_date': published_date,
            'modified_date': modified_date
         })
```

5 Create CVE nodes

[9]:

```
for val in tqdm(cve_clean_result):
    desc = ''
    for d in val['description']:
        desc = desc + " " + d.replace('\\','\\\\').replace('"', '\\"').
 ↪replace("'", "\\'")

    cql_create_node = f"""CREATE (:cve {{ cve_id: "{val['cve_id']}",
    description: "{desc}",
    cvss_base_severity: "{val['cvss_base_severity']}",
    cvss_base_score: {val['cvss_base_score']},
    published_date: {val['published_date'].split('-')[0]}
    }})"""
    run_query(cql_create_node)
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=152178.0),␣
 ↪HTML(value='')))
```

6 Create CVE to CWE relations

[10]:

```
for val in tqdm(cve_clean_result):
    cve_id = val['cve_id']
    for val2 in val['related_cwe_list']:
        cwe_id = val2
        cql_create_relationship = f"""MATCH (cve1:cve), (cwe1:cwe)
                          WHERE cve1.cve_id = '{cve_id}' AND cwe1.cwe_id =␣
 ↪'{cwe_id}'
```

```
                              CREATE (cve1)-[r:caused_by]->(cwe1)
                              RETURN type(r)"""
        run_query(cql_create_relationship)
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=152178.0),␣
 ↪HTML(value='')))
```

7 Create CVSS nodes

```
[11]: cvss_score_enum = ['CRITICAL', 'HIGH', 'MEDIUM', 'LOW', 'unknown']
      for val in cvss_score_enum:
          cql_create_node = f"""CREATE (:cvss {{ cvss_id: "{val}"}})"""
          run_query(cql_create_node)
```

8 Create CVSS relations to CVE

```
[20]: for val in tqdm(cve_clean_result):
          cve_id = val['cve_id']
          cvss_id = val['cvss_base_severity']

          cql_create_relationship = f"""MATCH (cve1:cve), (cvss1:cvss)
                                  WHERE cve1.cve_id = '{cve_id}' AND cvss1.cvss_id =␣
       ↪'{cvss_id}'
                                  CREATE (cve1)-[r:has_severity_of]->(cvss1)
                                  RETURN type(r)"""
          run_query(cql_create_relationship)
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=152178.0),␣
 ↪HTML(value='')))
```

9 Cleanup CPE company and product strings

```
[37]: company_set = set()
      product_set = set()

      for val in cve_clean_result:
          impacted_cpe_list = val['impacted_config']
          for val2 in impacted_cpe_list:
              company_set.add(val2.split(':')[3].replace('\\','').replace('"', '').
       ↪replace("'", "").replace("@", "").replace('+','_'))
              product_set.add(val2.split(':')[4].replace('\\','').replace('"', '').
       ↪replace("'", "").replace("@", "").replace('+','_'))
```

```
[32]: print(len(company_set))
```

```
22879
```

10 Create CPE company nodes

```
[39]: for val in tqdm(list(company_set)):
          cql_create_node = f"""CREATE (:cpe_comp {{ company_id: "{val}"}})"""
          run_query(cql_create_node)
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=22879.0),␣
 ↪HTML(value='')))
```

11 Creare CPE product nodes

```
[38]: for val in tqdm(list(product_set)):
          cql_create_node = f"""CREATE (:cpe_prod {{ product_id: "{val}"}})"""
          run_query(cql_create_node)
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=78994.0),␣
 ↪HTML(value='')))
```

12 Create CPE company and Product relation to CVE

```
[41]: for val in tqdm(cve_clean_result):
          cve_id = val['cve_id']
          impacted_cpe_list = val['impacted_config']
          company_already_connected = set()
          product_already_connected = set()
          for val2 in impacted_cpe_list:
              company = val2.split(':')[3]
              product = val2.split(':')[4]
              if company not in company_already_connected:
                  cql_create_relationship = f"""MATCH (cve1:cve), (cpe_comp1:cpe_comp)
                                WHERE cve1.cve_id = '{cve_id}' AND cpe_comp1.
       ↪company_id = '{company}'
                                CREATE (cve1)-[r:applies_to]->(cpe_comp1)
                                RETURN type(r);"""
                  run_query(cql_create_relationship)
                  company_already_connected.add(company)

              if product not in product_already_connected:
```

```
            cql_create_relationship = f"""MATCH (cve1:cve), (cpe_prod1:cpe_prod)
                          WHERE cve1.cve_id = '{cve_id}' AND cpe_prod1.
↪product_id = '{product}'
                          CREATE (cve1)-[r:applies_to]->(cpe_prod1)
                          RETURN type(r);"""
            run_query(cql_create_relationship)
            product_already_connected.add(product)
```

```
HBox(children=(HTML(value=''), FloatProgress(value=0.0, max=152178.0),␣
 ↪HTML(value='')))
```

```
[ ]:
```

CWE_unstructured_map

December 11, 2020

1 CWE Unstructured Map

```
[1]: import json
     import pandas as pd
     from openie import StanfordOpenIE
```

```
[2]: #load CWE impactnote,CWE description/extended description and CVE description␣
      ↪documents from Json files
```

```
[3]: with open('cwec_v4.2.json') as f:
         data = json.load(f)
```

```
[4]: weakness=data['Weakness_Catalog']['Weaknesses']['Weakness']
```

```
[ ]:
```

```
[5]: #CWE dis and ex_dis
     cwe_dis=[]
     cwe_ex_dis=[]
     for i in weakness:
         cwe_dis.append(('CWE-'+i['@ID'],i['Description']))
         if 'Extended_Description' in i:
             if type(i['Extended_Description'])==str:
                 cwe_ex_dis.append(('CWE-'+i['@ID'],i['Extended_Description']))
             if type(i['Extended_Description'])==dict:
                 if type(i['Extended_Description']['xhtml:p'])==str:
                     cwe_ex_dis.
      ↪append(('CWE-'+i['@ID'],i['Extended_Description']['xhtml:p']))
                 if type(i['Extended_Description']['xhtml:p'])==list:
                     for j in i['Extended_Description']['xhtml:p']:
                         cwe_ex_dis.append(('CWE-'+i['@ID'],j))
```

```
[6]: #CWE impact note
     cwe_impact_note=[]
     for i in weakness:
```

```
    if 'Common_Consequences' in i:
        j=i['Common_Consequences']['Consequence']
        if type(j)==list:
            for k in j:

                try:
                    if type(k['Note'])==list:
                        for l in k['Note']:


                            cwe_impact_note.append(('CWE-'+i['@ID'],l))
                    if type(k['Note'])==str:
                        cwe_impact_note.append(('CWE-'+i['@ID'],k['Note']))
                except:
                    next



        if type(j)==dict:
            if 'Note' in j:

                if type(j['Note'])==list:

                    for k in j['Note']:

                        cwe_impact_note.append(('CWE-'+i['@ID'],k))
                if type(j['Note'])==str:
                    cwe_impact_note.append(('CWE-'+i['@ID'],j['Note']))
```

[7]:
```
#CVE des
with open('nvdcve-1.1-2020.json') as f:
    data2 = json.load(f)
CVE=data2['CVE_Items']
```

[8]:
```
CVE_des=[]
for i in CVE:
#     print(i['cve']['CVE_data_meta']['ID'])
    for j in i['cve']['description']['description_data']:
        CVE_des.append(('CVE-'+i['cve']['CVE_data_meta']['ID'],j['value']))
#         for k in j['description']:
#             CVE_CWE.
 ↪append((i['cve']['CVE_data_meta']['ID'],i['cve']['description']['description_data'][0]['val
 ↪replace('"',"'").replace('\n',' '),k['value']))
```

[9]:
```
# check loaded ducuments
```

[10]:
```
cwe_dis[:3]
```

```
[10]: [('CWE-1004',
        'The software uses a cookie to store sensitive information, but the cookie is
      not marked with the HttpOnly flag.'),
       ('CWE-1007',
        'The software displays information or identifiers to a user, but the display
      mechanism does not make it easy for the user to distinguish between visually
      similar or identical glyphs (homoglyphs), which may cause the user to
      misinterpret a glyph and perform an unintended, insecure action.'),
       ('CWE-102',
        'The application uses multiple validation forms with the same name, which
      might cause the Struts Validator to validate a form that the programmer does not
      expect.')]
```

```
[11]: cwe_ex_dis[:3]
```

```
[11]: [('CWE-1004',
        "The HttpOnly flag directs compatible browsers to prevent client-side script
      from accessing cookies. Including the HttpOnly flag in the Set-Cookie HTTP
      response header helps mitigate the risk associated with Cross-Site Scripting
      (XSS) where an attacker's script code might attempt to read the contents of a
      cookie and exfiltrate information obtained. When set, browsers that support the
      flag will not reveal the contents of the cookie to a third party via client-side
      script executed via XSS."),
       ('CWE-1007',
        'Some glyphs, pictures, or icons can be semantically distinct to a program,
      while appearing very similar or identical to a human user. These are referred to
      as homoglyphs. For example, the lowercase "l" (ell) and uppercase "I" (eye) have
      different character codes, but these characters can be displayed in exactly the
      same way to a user, depending on the font. This can also occur between different
      character sets. For example, the Latin capital letter "A" and the Greek capital
      letter "A" (Alpha) are treated as distinct by programs, but may be displayed in
      exactly the same way to a user. Accent marks may also cause letters to appear
      very similar, such as the Latin capital letter grave mark "À" and its equivalent
      "À" with the acute accent.'),
       ('CWE-1007',
        'Adversaries can exploit this visual similarity for attacks such as phishing,
      e.g. by providing a link to an attacker-controlled hostname that looks like a
      hostname that the victim trusts. In a different use of homoglyphs, an adversary
      may create a back door username that is visually similar to the username of a
      regular user, which then makes it more difficult for a system administrator to
      detect the malicious username while reviewing logs.')]
```

```
[13]: cwe_impact_note[:3]
```

```
[13]: [('CWE-1004',
        'If the HttpOnly flag is not set, then sensitive information stored in the
      cookie may be exposed to unintended parties.'),
 ('CWE-1004',
  'If the cookie in question is an authentication cookie, then not setting the
HttpOnly flag may allow an adversary to steal authentication data (e.g., a
session ID) and assume the identity of the user.'),
 ('CWE-1007',
  "An attacker may ultimately redirect a user to a malicious website, by
deceiving the user into believing the URL they are accessing is a trusted
domain. However, the attack can also be used to forge log entries by using
homoglyphs in usernames. Homoglyph manipulations are often the first step
towards executing advanced attacks such as stealing a user's credentials, Cross-
Site Scripting (XSS), or log forgery. If an attacker redirects a user to a
malicious site, the attacker can mimic a trusted domain to steal account
credentials and perform actions on behalf of the user, without the user's
knowledge. Similarly, an attacker could create a username for a website that
contains homoglyph characters, making it difficult for an admin to review logs
and determine which users performed which actions.")]
```

[14]:
```
CVE_des[:3]
```

[14]:
```
[('CVE-CVE-2020-0001',
  'In getProcessRecordLocked of ActivityManagerService.java isolated apps are
not handled correctly. This could lead to local escalation of privilege with no
additional execution privileges needed. User interaction is not needed for
exploitation. Product: Android Versions: Android-8.0, Android-8.1, Android-9,
and Android-10 Android ID: A-140055304'),
 ('CVE-CVE-2020-0002',
  'In ih264d_init_decoder of ih264d_api.c, there is a possible out of bounds
write due to a use after free. This could lead to remote code execution with no
additional execution privileges needed. User interaction is needed for
exploitation Product: Android Versions: Android-8.0, Android-8.1, Android-9, and
Android-10 Android ID: A-142602711'),
 ('CVE-CVE-2020-0003',
  'In onCreate of InstallStart.java, there is a possible package validation
bypass due to a time-of-check time-of-use vulnerability. This could lead to
local escalation of privilege with no additional execution privileges needed.
User interaction is needed for exploitation. Product: Android Versions:
Android-8.0 Android ID: A-140195904')]
```

[15]:
```
cwe_dis=pd.DataFrame(cwe_dis,columns=['id','str'])
cwe_ex_dis=pd.DataFrame(cwe_ex_dis,columns=['id','str'])
cwe_impact_note=pd.DataFrame(cwe_impact_note,columns=['id','str'])
CVE_des=pd.DataFrame(CVE_des,columns=['id','str'])
```

1.1 auto phrase

[51]:

```
with open("cwe_impact_note.txt", 'w') as f:
    f.write("\n".join(list(cwe_impact_note['str'].drop_duplicates())).
 ↪replace('\n','').replace('\t',''))
```

[32]:

```
with open("cwe_dis.txt", 'w') as f:
    f.write("\n".join(list(cwe_dis['str'].drop_duplicates())).lower().
 ↪replace('\n','').replace('\t',''))
```

[36]:

```
with open("cwe_ex_dis.txt", 'w') as f:
    f.write("\n".join(list(cwe_ex_dis['str'].drop_duplicates())).lower().
 ↪replace('\n','').replace('\t',''))
```

[38]:

```
with open("CVE_des.txt", 'w') as f:
    f.write("\n".join(list(CVE_des['str'].drop_duplicates())).lower().
 ↪replace('\n','').replace('\t',''))
```

[ ]:

```
#run by auto phrase
```

[94]:

```
#read
with open("cwe_impact_note_AutoPhrase.txt", 'r') as f:
    lines = f.readlines()
cwe_impact_note_ap=  [line.replace('\n','').split('\t')[1] for line in lines]
```

[95]:

```
cwe_impact_note_ap[:5]
```

[95]:

```
['cross site scripting',
 'buffer overflow',
 'protection mechanisms',
 'execute arbitrary code',
 'data']
```

1.1.1 function for all string cat

[16]:

```
from spacy.lang.en import English
import spacy

#sentence tokenizer
nlp = English()
sbd = nlp.create_pipe('sentencizer')
nlp.add_pipe(sbd)

#nlp model
nlp_m = spacy.load("en_core_web_sm")
```

[17]:

```
#lemmatization
def lemmatize_text(text):
    text = nlp_m(text.lower())
    text = ' '.join([word.lemma_ if word.lemma_ != '-PRON-' else word.text for␣
 ↪word in text])
    return text
```

[18]:

```
# intial Rake for keyword extraction
from nlp_rake import Rake
rake = Rake(
    min_chars=3,
    max_words=3,
    min_freq=15,generated_stopwords_percentile=90
)
```

[19]:

```
# Function for extracting Keyword using RAKE after pre-processing (dedup,␣
 ↪lowercase, remove special charactors)

def make_key_words(df_string):
    x=' '.join(list(df_string['str'].drop_duplicates())).lower().
 ↪replace('\n','').replace('\t','').replace('*','').replace('/','').
 ↪replace('<','')
    return [a for a,b in rake.apply(x)]


# Tokenize document into list of sentence

def all_sentence(df_string):
    sents_list = []
    for i in df_string['str']:
        doc = nlp(i)
        for sent in doc.sents:
            sents_list.append(sent.text)
    return pd.DataFrame({'str':sents_list}).drop_duplicates()


# Among keywords, choose only Noun,Noun is determined by sampling from original␣
 ↪text

def detect_nn(sents_list,keywords):
    text = nlp_m(' '.join(list(sents_list['str'])))
    test=[]
    for wd in keywords:
        for token in text:
            if token.text==wd.split(' ')[-1]:
                test.append((wd,token.tag_,token.dep_))
    test=pd.DataFrame(test,columns=['kw','tag','dep'])
    a=test.groupby('kw')[['kw']].count()
    b=test[test["tag"]=='NN'].groupby('kw')[['tag']].count()
    test=pd.concat([a,b],axis=1)
    test['tag_p']=test['tag']/test['kw']
    test=test[test['tag_p']>0.5]
    return test


# Make triples using StanfordOpenIE by searching keyword sentences, and initial␣
 ↪clean within each sentence

def make_triple(test,sents_list):
    z=pd.DataFrame()

    with StanfordOpenIE() as client:

        for kw in test.index:
            x=[]
            y=[]
            for i in sents_list['str']:
                if kw in i:
                    x.append(i)
            x=pd.DataFrame({'str':x}).drop_duplicates()

            for text in x['str']:
                for triple in client.annotate(text):
                    y.
 ↪append((triple['subject'],triple['relation'],triple['object']))
            y=pd.DataFrame(y,columns=['e1','r','e2'])
            y=y[y['e1']==kw]
            y=triple_process(y)
            z=z.append(y)
    z=z[-z['r'].isin(['<','be','in'])]
    z=z[-(z['e1']==z['e2'])]
    return z.drop_duplicates()


# Post processing of triples: only keep verb relation, only keep longest phrase␣
 ↪as 2nd entities

def triple_process(triple_df):
    x=triple_df[-triple_df['r'].
 ↪isin(['of','could','may','can','be','to','in','will','on','at','by','than'])]
    x['r']=x['r'].str.replace(r'can|could|may', '').str.strip()
    x1=x['r'].str.split(' ')
    x2=x['e2'].str.split(' ')
    ct_r=[len(st) for st in x1]
    ct_e=[len(st) for st in x2]
    x['ct_r']=ct_r
    x['ct_e']=ct_e
    x=x[x.ct_r==1]
    return x.sort_values('ct_e', ascending=False).drop_duplicates(['e1','r']).
 ↪drop(['ct_r','ct_e'],axis=1)
```

1.2 Make triples -Method1 OPENIE

[31]:

```
# process for impact
```

[96]:

```
#lemmatization
cat=cwe_impact_note
cat['str']=cat['str'].apply(lemmatize_text)

cat.tail(3)
```

[96]:

```
         id                                               str
568  CWE-96  often the action perform by inject control cod…
569  CWE-98  the attacker may be able to specify arbitrary …
570  CWE-99  an attacker could gain access to or modify sen…
```

[97]:

```
#sentence tokenization
sents_list=all_sentence(cat)

sents_list.tail(3)
```

[97]:

```
                                                  str
824  alternatively , it may be possible to use norm…
825  an attacker could gain access to or modify sen…
826  this could allow access to protect file or dir…
```

[99]:

```
#keywords
keyword=list(set((make_key_words(cat)+cwe_impact_note_ap[:20])))

keyword[:12]
```

[99]:

```
['resource',
 'applications',
 'memory',
 'crash',
 'datum',
 'unauthorized',
 'logic',
 'modify',
 'program',
 'read',
 'software',
 'cross site scripting']
```

[100]:
```
#Noun keywords
NN=detect_nn(sents_list,keyword)

NN[:5]
```

[100]:
```
              kw    tag     tag_p
access       123   98.0  0.796748
application   68   68.0  1.000000
attack        68   67.0  0.985294
attacker     218  208.0  0.954128
case          21   21.0  1.000000
```

[36]:
```
# triples

final_cwe_impact_note=make_triple(NN,sents_list)

final_cwe_impact_note.head(10)
```

```
Starting server with command: java -Xmx8G -cp
/home/yupingph/stanfordnlp_resources/stanford-corenlp-full-2018-10-05/*
edu.stanford.nlp.pipeline.StanfordCoreNLPServer -port 9000 -timeout 60000
-threads 5 -maxCharLength 100000 -quiet True -serverProperties
corenlp_server-29386d5a973848da.props -preload openie
```

```
<ipython-input-19-89f358ea4cb7>:68: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['r']=x['r'].str.replace(r'can|could|may', '').str.strip()
<ipython-input-19-89f358ea4cb7>:73: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['ct_r']=ct_r
<ipython-input-19-89f358ea4cb7>:74: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['ct_e']=ct_e
```

[36]:

```
                e1             r                                              e2
280  application        handle  response from untrusted application on device
199  application           has               expectation for content of state
261  application        launch                                        activity
270  application          read                                           datum
22        attack         steal                            cross site scripting
906       attack  identifiable                           other important datum
16        attack         forge                                       log entry
251       attack        inject                                 arbitrary reply
285       attack       execute                                   other command
855       attack         allow                                  malicious host
```

[37]:

```python
final_cwe_impact_note[final_cwe_impact_note.e1=='attacker'].head(10)
```

[37]:

```
              e1         r                                                 e2
799   attacker   violate    application 's expectation for content of state
642   attacker      gain             access to user account by user account
82    attacker    render            file unusable by corrupt format of file
134   attacker  leverage  additional information provide by default erro…
1116  attacker    bypass                  web browser 's same origin policy
168   attacker  identify            exploitable vulnerability in one device
117   attacker    modify               single byte arbitrary code execution
448   attacker     guess            gain access to restricted functionality
58    attacker    insert                          false entry into log file
776   attacker  retrieve       legitimate user 's authentication credential
```

[122]:

```python
# process for cwe des
cat=cwe_dis
cat['str']=cat['str'].apply(lemmatize_text)

sents_list=all_sentence(cat)

keyword=make_key_words(cat)

NN=detect_nn(sents_list,keyword)

final_cwe_dis=make_triple(NN,sents_list)
```

```
Starting server with command: java -Xmx8G -cp
/home/yupingph/stanfordnlp_resources/stanford-corenlp-full-2018-10-05/*
edu.stanford.nlp.pipeline.StanfordCoreNLPServer -port 9000 -timeout 60000
-threads 5 -maxCharLength 100000 -quiet True -serverProperties
corenlp_server-0c0b6a14f2304cbd.props -preload openie
```

```
<ipython-input-119-3e93869712b4>:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['r']=x['r'].str.replace(r'can|could|may', '').str.strip()
<ipython-input-119-3e93869712b4>:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['ct_r']=ct_r
<ipython-input-119-3e93869712b4>:9: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['ct_e']=ct_e
```

[128]:

```
final_cwe_dis
```

[128]:

```
                         e1             r  \
37                   access       contain
106                   actor     determine
185                   actor       perform
260             application        expose
37              application      truncate
86              application          file
25              application           has
90              application         under
114             application       contain
201             application       perform
193             application           use
20              application       prevent
98              application     determine
287                attacker         cause
286                attacker       perform
98                 attacker       inflict
141                attacker        upload
118                attacker        inject
342                attacker        modify
48                 attacker         exist
117                attacker        ignore
310                attacker       control
116                attacker        induce
168                attacker      traverse
318                attacker     reference
315                attacker          gain
```

337	attacker	take
30	attacker	put
195	attacker	read
38	attacker	execute
97	attacker	trick
61	attacker	try
57	attacker	bypass
42	attacker	set
20	class	contain
9	class	have
265	code	have
3	code	contain
247	code	constant
45	code	preserve
32	code	use
211	code	include
190	datum	serialize
45	entity	has
232	file	contain
51	file	leave
66	file	has
169	function	return
195	function	influence
54	information	determine
211	information	observe
252	information	log
327	information	change
131	input	have
19	mechanism	recognize
23	method	contain
68	method	should
105	method	read
27	method	support
79	method	process
123	process	allow
286	product	use
154	product	prevent
185	product	manage
268	product	inherit
343	product	generate
226	product	provide
239	product	make
168	product	implement
436	product	specify
34	product	enable
409	product	determine
265	product	define

```
76                  product           has
79                  product         allow
373                 product          embe
396                 product     encounter
405                 product       perform
140                 product        ensure
219                 product        expose
17                  product          have
217                 product     calculate
209                 product      subtract
378                 product       contain
387                 product          call
359                 product       compare
139                 product      validate
254                 product        assign
130                 product       receive
304                 product        divide
318                 product          find
324                 product        access
330                 product        within
133                 program          omit
192                 program       violate
234                 program        obtain
95                  program          send
179                 program          call
191                 program       declare
200                 program       compare
14                  program          copy
120                 program        define
85                  program       contain
231                 program   dereference
79                  program           use
233                 program       convert
237                 program         check
31                 resource          make
23                   result       perform
29                 security       contain
149   sensitive information        change
254                software          make
82                 software           use
809                software       contain
19                 software        create
121                software  misinterpret
959                software    neutralize
745                software          call
829                software      allocate
430                software       perform
```

```
226                software       specify
549                software    initialize
153                software        permit
432                software        verify
429                software      transmit
348                software           set
442                software         leave
445                software           has
463                software         check
934                software         limit
1003               software       receive
687                software        modify
800                software        define
854                software          omit
68                 software        manage
1002               software     establish
794                software       declare
766                software         allow
29                 software          have
449                software         treat
294                software      generate
297                software      identify
214                software        decode
553                software        remove
920                software       include
839                software        impose
887                software          read
872                software         write
143                software          save
765                software        access
353                software      preserve
408                software      validate
514                software      restrict
11          software system         allow
20                   system     implement
107                  system        create
221                    user     influence
95                     user         sniff
117                    user   impersonate
122                    user           use
164                    user           has
223                    user          know
3                      user  misinterpret
101              validation         allow
58                 variable       contain
67                 variable           has
34                 weakness       amplify
```

```
                                                     e2
37                            sensitive information
106                     file 's existence otherwise
185                                          action
260                remote interface for entity bean
37                           processing of security
86                        system content disclosure
25                                   model of state
90                                   direct control
114                                            code
201                                        security
193                                        getlogin
20                                         attacker
98                                             size
287           software operate on unauthorized file
286         unauthorized action against target file
98                           damage to their system
141                          file of dangerous type
118                                window unc share
342                                  command of xml
48                              protection to asset
117                           other error condition
310                              structure of query
116                   unexpected behavior unnoticed
168                                     file system
318                                   arbitrary dtd
315                                       privilege
337                                       advantage
30                                           system
195                                         content
38                                            datum
97                                             user
61                                             keep
57                                       validation
42                                           system
20              unnecessarily large number of child
9                                 inheritance level
265           return statement inside finally block
3                          callable control element
247                                  critical value
45                           associated information
32                              data representation
211                                           virus
190                                           class
45                                  right over time
232       sensitive information pertain to application
```

51	function
66	existence
169	address of stack variable
195	its state
54	value of reference token
211	system behavior
252	file
327	datum
131	property
19	complex datum input
23	large number of datum access
68	when should use instead
105	sensitive information
27	comparison
79	request
123	temporary access to shared resource
286	hash algorithm more efficiently than brute for…
154	revealing of unnecessary system information wi…
185	invalid use of special element in user
268	set of insecure permission for object
343	core dump file in directory resource
226	different response to incoming request
239	resource available to unauthorized actor
168	insufficient protection against information le…
436	permission for security critical resource
34	direct memory access capable device
409	how much memory allocate
265	set of insecure permission
76	architecture mirror region ensure
79	address region
373	incorrect document
396	error condition
405	same operation
140	index reference
219	sensitive information
17	attack surface
217	minimum value
209	one value
378	assert
387	routine
359	string
139	index
254	privilege
130	input
304	value
318	resource
324	resource

330	environment
133	cause code associate with multiple condition e…
192	secure code principle for mobile code
234	value from untrusted source
95	non cloned mutable datum
179	thread 's run
191	array public
200	object reference
14	input buffer
120	signal handler
85	code sequence
231	result
79	chroot
233	value
237	value
31	them easy target for attacker
23	cryptographic operation
29	semiconductor defect
149	datum
254	invalid assumption how protocol datum memory b…
82	unnecessarily complex internal representation …
809	conditional statement with multiple logical ex…
19	immutable text string use string concatenation…
121	whether from attacker in security relevant fas…
959	user control input for alternate script syntax
745	non reentrant function in concurrent context
829	reusable resource of resource
430	key exchange with actor
226	regular expression in way
549	datum store use input
153	unauthorized modification of memory
432	identity of actor
429	sensitive critical datum
348	permission of object
442	pattern of value
445	random number generator
463	state of resource
934	number of time
1003	message from endpoint
687	ssl context
800	public method
854	important detail
68	data access
1002	communication channel
794	critical variable
766	user input
29	loop condition

```
449                                     untrusted datum
294                                       error message
297                                     error condition
214                                          same input
553                                  temporary resource
920                                   web functionality
839                                         restriction
887                                              buffer
872                                               datum
143                                                user
765                                            resource
353                                          permission
408                                         certificate
514                                       functionality
11                                             attacker
20               security token mechanism differentiate
107                             insecure temporary file
221                         name of variable at runtime
95                                      network traffic
117                                          trust user
122                                      target machine
164                                   explicit approval
223                                   original password
3                                                 glyph
101                                            attacker
58            sensitive information about remote server
67                                                value
34                        consequence of other weakness
```

[124]:

```
# process for cwe expanded des

cat=cwe_ex_dis

cat['str']=cat['str'].apply(lemmatize_text)
sents_list=all_sentence(cat)

keyword=make_key_words(cat)

NN=detect_nn(sents_list,keyword)

final_cwe_ex_dis=make_triple(NN,sents_list)
```

```
Starting server with command: java -Xmx8G -cp
/home/yupingph/stanfordnlp_resources/stanford-corenlp-full-2018-10-05/*
edu.stanford.nlp.pipeline.StanfordCoreNLPServer -port 9000 -timeout 60000
-threads 5 -maxCharLength 100000 -quiet True -serverProperties
corenlp_server-90002ef59e234dce.props -preload openie
```

```
<ipython-input-119-3e93869712b4>:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['r']=x['r'].str.replace(r'can|could|may', '').str.strip()
<ipython-input-119-3e93869712b4>:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['ct_r']=ct_r
<ipython-input-119-3e93869712b4>:9: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['ct_e']=ct_e
```

[125]:
```
final_cwe_ex_dis
```

[125]:
```
                          e1             r  \
863                   access          list
268                   access     integrate
292                   access       contain
559                   access     whereupon
1327                  access       include
289                   action           use
158                   action        access
261                   action         occur
418                   action          omit
111                    agent         write
162                    agent          gain
38                     agent        create
68               application           use
491              application     determine
510              application         allow
67               application        obtain
417              application       display
532              application       protect
378              application          echo
375              application        expose
366              application          make
21               application       produce
```

309	application	store
307	application	handle
183	application	locate
517	application	contain
508	application	want
449	application	perform
392	application	have
380	application	has
31	application	give
32	application	execute
281	application	control
286	application	improve
301	application	pass
70	asset	should
83	asset	assume
119	asset	have
1752	attack	allow
1444	attack	extend
1775	attack	change
1780	attack	cause
1756	attack	set
205	attack	compromise
1301	attack	product
1754	attack	execute
1175	attacker	hide
1349	attacker	transfer
556	attacker	source
833	attacker	consume
496	attacker	send
1237	attacker	compute
1518	attacker	has
1355	attacker	compromise
1388	attacker	bypass
1192	attacker	explore
748	attacker	make
464	attacker	infer
989	attacker	mount
434	attacker	use
527	attacker	examine
1410	attacker	supply
782	attacker	perform
1422	attacker	influence
845	attacker	execute
551	attacker	sniff
154	attacker	extract
1109	attacker	conduct
1096	attacker	gain

1471	attacker	have
1360	attacker	encode
389	attacker	modify
1485	attacker	manipulate
1294	attacker	cause
896	attacker	introduce
932	attacker	understand
1058	attacker	obtain
685	attacker	reverse
1060	attacker	map
1068	attacker	assume
568	attacker	create
1493	attacker	change
143	attacker	alter
993	attacker	launch
1030	attacker	read
1019	attacker	steal
1596	attacker	specify
1038	attacker	spoof
1570	attacker	establish
883	attacker	select
868	attacker	display
1579	attacker	inject
1061	attacker	escalate
1484	attacker	affect
1206	attacker	access
1220	attacker	trigger
1231	attacker	exploit
1265	attacker	contain
1414	attacker	control
1476	attacker	stuff
430	attacker	simplify
708	attacker	guess
133	attacker	forge
543	attacker	learn
713	attacker	determine
552	attacker	see
702	attacker	provide
162	attacker	populate
32	attacker	craft
45	attacker	trick
575	attacker	possess
482	attacker	traverse
302	attacker	decipher
1402	attacker	within
282	attacker	find
383	attacker	disable

1540	attacker	delete
22	attacker	redirect
1449	attacker	invoke
723	attacker	nee
716	attacker	test
8	attacker	run
781	attacker	win
888	attacker	upload
871	attacker	convince
875	attacker	initialize
695	attacker	break
1164	attacker	request
537	attacker	force
1120	attacker	deny
687	attacker	impersonate
557	attacker	circumvent
554	attacker	locate
308	authentication	have
115	certificate	follow
133	certificate	ensure
134	certificate	must
97	certificate	wield
132	certificate	has
192	class	become
145	class	introduce
187	class	access
14	client	ignore
12	client	skip
203	code	read
1035	code	have
530	code	make
140	code	include
995	code	perform
132	code	implement
857	code	misclassify
9	code	compare
98	code	become
311	code	specify
208	code	execute
65	command	resende
68	command	change
76	component	have
49	component	compromise
140	component	use
16	component	interpret
55	component	master
146	component	decode

165	component	misinterpret
483	control	should
13	cookie	read
32	cookie	contain
490	datum	monitor
1009	datum	cause
358	datum	contain
678	datum	cross
954	datum	modify
51	datum	allow
60	datum	enter
216	datum	overwriting
2	design	handle
128	developer	use
65	developer	maintain
171	developer	create
69	developer	reduce
48	developer	update
146	developer	introduce
137	developer	insert
95	developer	protect
71	developer	choose
108	developer	handle
122	developer	release
140	developer	code
148	developer	want
173	developer	assume
307	device	have
276	device	endure
254	device	allow
571	device	through
16	device	has
344	device	employ
400	device	enter
275	device	become
282	device	support
434	device	about
19	entity	obtain
322	entity	have
274	error	have
106	error	cause
156	error	overlap
357	file	contain
432	file	have
630	file	perform
415	function	make
280	function	become

327	function	has
568	function	handle
229	function	alter
261	function	follow
507	function	include
167	function	get
461	function	indicate
178	function	fail
571	function	use
130	functionality	contain
233	hardware	switch
234	hardware	use
121	hash	reduce
40	implementation	attack
225	implementation	allow
125	index	has
319	information	include
746	information	change
639	information	bypass
185	information	cause
694	information	enable
214	information	make
272	information	lower
261	information	contain
101	input	follow
368	input	enter
69	issue	prevent
116	issue	make
237	issue	have
233	issue	ignore
89	issue	suggest
115	issue	maintain
16	language	allow
247	language	contain
248	lock	include
71	lock	get
73	lock	become
326	lock	has
26	mechanism	include
42	mechanism	scale
308	memory	prevent
16	message	contain
75	method	throw
52	method	isolate
62	method	have
37	modification	allow
189	object	contain

125	object	have
94	object	save
133	object	has
248	part	make
212	password	gain
341	password	compute
339	password	compare
41	place	exploit
73	pointer	give
169	pointer	contain
195	pointer	read
240	process	combine
306	process	elevate
286	process	open
501	product	define
307	product	implement
489	product	inherit
88	product	produce
150	product	provide
151	product	manage
488	product	enter
468	product	use
367	product	behave
483	product	make
467	product	find
410	product	about
513	product	prepare
309	product	exit
396	product	has
469	product	from
426	product	log
419	product	expect
707	program	modify
697	program	has
346	program	manipulate
689	program	violate
709	program	create
434	program	crash
696	program	change
666	program	return
869	program	use
765	program	give
652	program	line
378	program	lock
570	program	call
470	program	remove
450	program	patch

405	program	recover
99	programmer	have
21	programmer	leave
124	programmer	begin
48	programmer	remedy
119	programmer	trust
20	programmer	provide
1	programmer	cause
18	programmer	accept
144	programmer	must
169	programmer	use
131	programmer	has
104	programmer	catch
32	programmer	from
31	programmer	perform
23	programmer	avoid
5	programmer	fix
0	request	contain
207	request	provide
728	resource	should
527	resource	have
325	resource	include
731	resource	issue
630	resource	program
787	resource	use
18	resource	support
792	resource	require
734	resource	lead
370	resource	apply
232	result	violate
197	result	control
513	security	depend
707	security	allow
790	security	determine
263	security	restrict
85	sensitive information	include
59	sensitive information	cause
66	sensitive information	make
141	server	store
190	server	handle
279	server	make
41	server	request
48	server	use
255	server	bypass
50	server	sniff
257	server	send
45	service	allow

62	signal handler	interrupt
21	signal handler	have
124	soc	measure
42	software	perform
553	software	have
291	software	define
318	software	trust
696	software	use
528	software	restrict
709	software	allow
572	software	unlock
518	software	operate
96	software	retain
561	software	lock
138	software	follow
678	software	has
377	software	generate
90	software	cause
651	software	intend
582	software	during
669	software	remove
274	software	modify
525	software	take
350	software	connect
379	software	require
524	software	choose
749	system	need
321	system	implement
594	system	has
612	system	combine
804	system	generate
807	system	send
755	system	should
156	system	utilize
335	system	employ
327	system	remain
338	system	sleep
784	system	use
11	technique	employ
60	technique	break
117	transaction	require
414	trust	traverse
558	trust	know
564	trust	afford
202	type	username
557	user	gain
359	user	send

```
476                    user           set
288                    user      download
50                     user          read
18                     user         click
742                    user        bypass
161                    user          list
335                    user        enable
170                    user        launch
725                    user       perform
656                    user        access
797                    user          have
194                    user        change
193                    user         avoid
203                    user      generate
778                    user           has
714                    user          into
163                    user           see
478                    user        modify
255                    user       receive
261                    user    compromise
341                    user       display
129           vulnerability          gain
330                weakness          lead
439                weakness         allow
45                 weakness        launch
14                 weakness      indicate
319                weakness         cover
207                weakness        affect
135                weakness          turn
353                weakness          take
446                weakness         cause

                                                     e2
863                                   process on system
268                                     hardware engine
292                               sensitive information
559                                     untrusted agent
1327                                        other datum
289                         trust level of other domain
158                                      asset for read
261                                        user proceed
418                                                step
111                                  access to resource
162                                     access to asset
38                                          memory alia
68            container 's resource management facility
491                             proper course of action
```

```
510              access to http://www.example.com/mypage
67                                connection to resource
417                                     name of resource
532                                      token parameter
378                                        error message
375                                         file content
366                                     outgoing request
21                                          debug binary
309                                       sensitive file
307                                        error message
183                                    critical resource
517                                     certain function
508                                                allow
449                                       authentication
392                                            mechanism
380                                            structure
31                                            capability
32                                               command
281                                               access
286                                           efficiency
301                                          information
70                                     should accessible
83                                             immutable
119                                               access
1752    modification of critical program state variable
1444                                     scope of damage
1775                                   state of software
1780                                   instable behavior
1756                                    program variable
205                                   security guarantee
1301                                              itself
1754                                                code
1175  information about file from system at command …
1349  private information from victim 's machine to …
556     routing be disable across much of internet today
833         more resource than their access level permit
496                          sized input to > > operator
1237          hash value use dictionary attack technique
1518          workload for brute force password cracking
1355                       victim 's account on web site
1388            intended protection of captcha challenge
1192                     web site look for vulnerability
748                  unintentional request to web server
464                   certain property about private key
989                             denial of service attack
434                  knowledge of internal operation
527                               content of memory dump
```

1410	pointer for memory location
782	large number of attack
1422	portion of uninitialized memory
845	access unauthorized datum file
551	traffic between victim machine
154	some of original information
1109	other attack against user
1096	unauthorized access to system
1471	knowledge of original password
1360	malicious portion of attack
389	externally control format string
1485	business logic of software
1294	denial of service
896	new malicious behavior
932	logic of script
1058	portion of document
685	engineer binary code
1060	application 's structure
1068	privilege of process
568	certificate with name
1493	behavior of command
143	intend control flow
993	further attack
1030	source code
1019	user credential
1596	local file
1038	symbolic link
1570	communication channel
883	unexpected class
868	erroneous datum
1579	additional logic
1061	their privilege
1484	count separately
1206	system file
1220	unusual condition
1231	weak algorithm
1265	unexpected value
1414	the offset
1476	ballot box
430	their attack
708	next value
133	log entry
543	bad yet
713	random number
552	response packet
702	alternate password
162	data field

32	malicious input
45	autonomous vehicle
575	correct key
482	file system
302	vulnerability
1402	input
282	input
383	protection
1540	datum
22	user
1449	function
723	consider
716	predict
8	script
781	race
888	file
871	user
875	variable
695	compromise
1164	device
537	function
1120	service
687	actor
557	requirement
554	themselves
308	weakness
115	chain of trust
133	datum integrity
134	must valid
97	resource
132	host
192	peer class in bytecode
145	several security concern
187	private field
14	authentication failure
12	authentication
203	variable amount of datum
1035	access to local dom
530	change to datum send
140	dead code
995	other attack
132	intended behavior
857	supply file
9	reference
98	obsolete
311	quantity
208	it

65	same command
68	them
76	same view of overall system
49	soc boot firmware
140	decoding method
16	datum
55	transaction
146	data
165	output
483	should pass
13	sensitive information
32	sensitive datum
490	timing of operation
1009	denial of service
358	sensitive information
678	trust boundary
954	xml syntax
51	attacker
60	application
216	possible
2	data access operation
128	servlet member field
65	list of ban
171	command use interpolation
69	risk of vulnerability
48	validation logic
146	related weakness
137	malicious code
95	their product
71	default value
108	information
122	memory
140	program
148	initialize
173	that
307	improperly secure power management feature
276	limited number of write
254	device configuration control
571	memory card port
16	internal information
344	many power
400	osat facility
275	unreliable
282	feature
434	operation
19	access to ip
322	obvious risk

274	security consequence
106	null
156	cause
357	sensitive information
432	long name
630	parsing
415	certain assumption about datum
280	programming language evolve
327	stack frame
568	multiple signal
229	mutable datum
261	search order
507	realpath other
167	get unsafe
461	error status
178	value
571	state
130	grant access to additional functionality
233	corrupt
234	datum
121	load
40	result
225	authentication
125	result
319	otherwise useful in further exploitation
746	behavior of system
639	intended security policy
185	crash
694	attacker
214	leak
272	security
261	easy
101	particular syntax
368	control plane
69	software from run reliably e.g. by trigger
116	it more difficult to port
237	have report for asp
233	critical file
89	poor encapsulation
115	software
16	direct addressing of memory location
247	issue
248	exclusive lock
71	clear
73	programmable
326	implementation
26	encapsulation failure

42	device voltage
308	modification
16	command
75	generic form of exception defeat
52	low bit of value
62	have call
37	access of configuration information
189	reference to particular resource
125	equal hashcode
94	them
133	finalize
248	assumption about content of field
212	privilege associate
341	its hash
339	it
41	latent vulnerability
73	even give set to null
169	reference to arbitrary memory location
195	datum
240	multiple independent component
306	their privilege
286	file
501	its control sphere within code itself
307	hardware base access control check
489	weakness associate with state
88	new kind of resource
150	web base application
151	underlie operating system
488	less secure state
468	directory search path
367	differently base
483	it compromise
467	executable library
410	internal operation
513	control message
309	manufacturing stage
396	vendor
469	system
426	event
419	uphold
707	ssl _ ctx object
697	memory management datum structure
346	device clock frequency
689	follow ejb guideline
709	ssl object
434	sensitive information
696	value store

666	pointer
869	pointer
765	attacker
652	trust
378	cpu
570	free
470	one
450	software
405	information
99	choice of several different mechanism
21	door open for attacker
124	new thread of control
48	password management problem
119	unvalidated datum
20	unexpected input
1	unexpected result
18	xml document
144	must careful
169	entity encoding
131	assumption
104	nullpointerexception
32	view
31	conversion
23	use
5	weakness
0	lf line feed
207	malicious content
728	for weakness should still valid upon subsequen…
527	explicit instruction how be create
325	database connection pool entry
731	similar to cwe-825
630	sensitive user datum
787	incompatible type
18	different feature
792	initialization
734	issue
370	authorization
232	assumption make by program
197	looping
513	how be use within code
707	enterprise bean
790	resource usage
263	assignment
85	otherwise useful in further exploitation
59	crash
66	leak
141	set of file

190	multiple simultaneous connection
279	cross domain request
41	authentication information
48	authentication information
255	access control
50	traffic
257	request
45	external control of system setting
62	normal functionality of program
21	have set
124	code
42	intend central data manager may have be explic…
553	fix list of special character
291	isolated memory region policy
318	integrity of information source
696	too much power
528	access to file
709	user 's input
572	critical resource
518	software assume
96	other resource
561	critical resource
138	certain protocol
678	attack surface
377	predictable value
90	software hang
651	send
582	execution
669	javascript
274	index
525	base
350	believe
379	unpredictability
524	action
749	guard by complex security check
321	multiple level of policy
594	reuse of free memory
612	multiple independent component
804	new temporary password
807	original password
755	should prefer
156	web front
335	many power
327	oem forget
338	state
784	sequential
11	flip

```
60                                            algorithm
117                                             execute
414                                      several entity
558                                               class
564                                                know
202                                              enable
557                    access to their account in event
359                                 object of same name
476                                primary key to value
288                            file from unknown source
50                       datum use spreadsheet software
18                                link to external site
742                            intended security policy
161                          information about process
335                              security option enable
170                             attack against software
725                                      certain action
656                                          other file
797                            administrative privilege
194                                      their password
193                                      reuse password
203                                       poor password
778                                               input
714                                             browser
163                                         information
478                                                 key
255                                         certificate
261                                            protocol
341                                         information
129                        well understanding of system
330                 depend on behavior of expose method
439                           can influence by attacker
45                             denial of service attack
14                                cycle between package
319                            three distinct situation
207                                 program correctness
135                                          they occur
353                                        several form
446                                downstream component
```

[126]:

```
# process for CVE

cat=CVE_des.head(1500)
cat['str']=cat['str'].apply(lemmatize_text)

sents_list=all_sentence(cat)
```

```
keyword=make_key_words(cat)

NN=detect_nn(sents_list,keyword)


final_CVE_des=make_triple(NN,sents_list)
```

```
<ipython-input-126-100a0795a657>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  cat['str']=cat['str'].apply(lemmatize_text)
```

```
Starting server with command: java -Xmx8G -cp
/home/yupingph/stanfordnlp_resources/stanford-corenlp-full-2018-10-05/*
edu.stanford.nlp.pipeline.StanfordCoreNLPServer -port 9000 -timeout 60000
-threads 5 -maxCharLength 100000 -quiet True -serverProperties
corenlp_server-4cf8ac694ccd4e86.props -preload openie
```

```
<ipython-input-119-3e93869712b4>:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['r']=x['r'].str.replace(r'can|could|may', '').str.strip()
<ipython-input-119-3e93869712b4>:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['ct_r']=ct_r
<ipython-input-119-3e93869712b4>:9: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  x['ct_e']=ct_e
```

[ ]:

```

```

[127]:

```
final_CVE_des
```

```
[127]:                                e1            r  \
       2225                       access       reveal
       2398                       access       plugin
       2636                       access       upload
       189                   application       depend
       0                     application    construct
       2                     application         make
       186                   application      respond
       206                   application        allow
       344                      attacker         view
       822                      attacker      achieve
       981                      attacker         give
       1560                     attacker      exploit
       281                      attacker          run
       258                      attacker      require
       831                      attacker      perform
       1115                     attacker       inject
       75                       attacker       bypass
       1812                     attacker         read
       861                      attacker          use
       608                      attacker      request
       1460                     attacker     discover
       951                      attacker         gain
       290                      attacker         send
       1647                     attacker     initiate
       1603                     attacker        cause
       1537                     attacker        crash
       1760                     attacker       obtain
       1101                     attacker          set
       798                      attacker      execute
       816                      attacker      conduct
       918                      attacker         take
       1263                     attacker          add
       1286                     attacker         post
       1005                     attacker     retreive
       538                      attacker         sign
       476                      attacker       browse
       1709                     attacker        alter
       1642                     attacker       delete
       1663                     attacker        write
       1680                     attacker      elevate
       514                      attacker    formulate
       1786                     attacker    establish
       1777                     attacker       hijack
       1753                     attacker       modify
       1260                     attacker       change
       1677                     attacker       access
```

1389	attacker	manipulate
1419	attacker	provide
1597	attacker	disclose
1290	attacker	create
1325	attacker	edit
434	attacker	include
586	attacker	impersonate
892	attacker	get
871	attacker	activate
417	attacker	replay
604	attacker	retrieve
1037	attacker	generate
595	attacker	win
1609	attacker	replace
1566	attacker	plant
1802	attacker	craft
1820	attacker	leverage
1749	attacker	trick
1737	attacker	overwrite
950	attacker	interact
944	attacker	compute
1064	attacker	download
668	attacker	insert
609	attacker	authorize
1345	attacker	close
1354	attacker	approve
1357	attacker	disapprove
964	authenticate user	achieve
330	authenticate user	enable
979	authenticate user	execute
1010	authenticate user	conduct
1023	authenticate user	exploit
1031	authenticate user	trigger
198	buffer overflow	involve
15	buffer overflow	allow
150	buffer overflow	enable
295	bypass	intend
18	default	assist
73	elevation	exploit
287	elevation	handle
411	elevation	connect
32	exploitation	require
29	exploitation	escalate
767	file	contain
587	file	use
520	flaw	allow
157	flaw	bypass

```
135                          flaw       affect
410                          flaw      selinux
422                          flaw          use
147                          flaw       obtain
149                          flaw      perform
237                          flaw        occur
253     improper input validation        allow
5                           issue         have
86                          issue       affect
54                 local attacker      exploit
94                 local attacker         read
108                local attacker      perform
89                 local attacker       obtain
119                local attacker        cause
83                 local attacker       modify
5                  local attacker      include
19                 local attacker       inject
62                 local attacker        plant
35                 local attacker       bypass
79                 local attacker          use
0     microsoft sharepoint server       handle
161               privileged user       enable
29                 race condition        allow
82                 race condition    reference
211               remote attacker     discover
278               remote attacker        cause
160               remote attacker         gain
236               remote attacker        crash
130               remote attacker      execute
269               remote attacker     disclose
190               remote attacker      achieve
173               remote attacker     generate
37                remote attacker       inject
186               remote attacker     download
1                 remote attacker       replay
125                       request      include
1236                      service       create
1392                      service        allow
981                       service       handle
1343                      service        bring
1515                       system       enable
968                        system          has
281          unauthenticated user       enable
3028                         user        cause
798                          user       enable
2440                         user         view
2906                         user       create
```

```
2443                         user        guess
2759                         user      achieve
2809                         user       escape
2682                         user       change
2532                         user       upload
3017                         user        abuse
2502                         user       handle
2747                         user          run
3009                         user          use
2951                         user          put
3095                         user      perform
1491                vulnerability       affect
1534                vulnerability        allow
628                 vulnerability         take
1379                vulnerability         send
614                 vulnerability          run
319                 vulnerability        place
238                        window       handle
594                        window       update
590                        window      connect
709                        window        stack

                                                     e2
2225                           confidential information
2398                                      functionality
2636                                               file
189   input supply be recognize as associate with va…
0                                   type _ toast window
2                                      window clickable
186                                              depend
206                                              access
344   aka microsoft yourphone application for androi…
822   cross tenant virtual machine access by corrupt…
981   lack of authentication capability in such vers…
1560  incorrect permission set by affected pi system…
281   aka window alpc elevation of privilege vulnera…
258   aka windows installer elevation of privilege v…
831           brute force calculation of encryption key
1115                    untrusted input inside csv file
75             passcode requirement of app.the security
1812                     content of artemis shadow file
861                        mi _ console command cascade
608                 list of their authorize application
1460                         reference username via api
951                     access to sensitive information
290              specially craft authentication request
1647                         password change for device
```

1603	denial of service condition
1537	pi network manager service
1760	signature of protect pointer
1101	second cookie with name
798	arbitrary os command
816	privilege escalation attack
918	control of robot
1263	new news article
1286	comment on article
1005	which will allow
538	cab archive use
476	browser cache content
1709	ansible _ fact
1642	file outside webaccess
1663	endless log statement
1680	their privilege level.to
514	more precise attack
1786	access to system
1777	grub verification process
1753	iscsi configuration
1260	global setting
1677	local machine
1389	arbitrary file
1419	javascript code
1597	sensitive information
1290	custom field
1325	glossary term
434	arbitrary command
586	bluetooth br
892	root shell
871	failsafe mode
417	authentication traffic
604	client secret
1037	cpu activity
595	control script
1609	them
1566	binary
1802	url
1820	issue
1749	user
1737	file
950	exfiltrate
944	token
1064	file
668	javascript
609	application
1345	ticket

```
1354                                            comment
1357                                            comment
964                          sql injection in app model
330                  escalation of privilege of service
979                                      arbitrary code
1010                                         xss attack
1023                                               flaw
1031                                                use
198          supplemental prompting by kerberos library
15                              enable via local access
150                                            attacker
295                      restriction on message reading
18                                                  app
73    aka microsoft windows elevation of privilege v…
287   aka microsoft splwow64 elevation of privilege …
411                                     user experience
32              knowledge of service name of target pod
29     privilege in unauthorized information disclosure
767                                            php code
587                                                 dot
520                  local attacker with user privilege
157                        gain access to workspace pod
135                        ansible engine version 2.7.x
410                             lsm hook implementation
422                                   leaked oauthtoken
147                               sensitive information
149                                          xss attack
237                                               while
253               potentially enable via network access
5      have fix in 2.28.0 with improved memory handling
86                                         chip produce
54    incorrect permission set by affected pi system…
94                       content of artemis shadow file
108                             spectre v2 style attack
89                         signature of protect pointer
119                                   denial of service
83                                  iscsi configuration
5                                     arbitrary command
19                                    arbitrary command
62                                               binary
35                                             password
79                                                 flaw
0                                oauth token validation
161                  denial of service via local access
29                        potentially enable via access
82                                  free too early time
211                          reference username via api
```

```
278                         denial of service condition
160                     access to sensitive information
236                                pi archive subsystem
130                                arbitrary os command
269                               sensitive information
190                                      code execution
173                                        cpu activity
37                                 arbitrary javascript
186                                                file
1                                                   otp
125          additional datum in allpopupdata parameter
1236                   server side request forgery risk
1392                   arbitrary writing to file system
981                                              object
1343                                        manipulator
1515                                  targetclid socket
968                                            hostname
281                  escalation of privilege of service
3028           stack overflow lead to denial of service
798                  denial of service via local access
2440                   ticket customer detail associate
2906                                  new admin account
2443                                   valid user email
2759                                        root access
2809                             restricted environment
2682                                               them
2532                                               file
3017                                               flaw
2502                                             object
2747                                            command
3009                                               flaw
2951                                        application
3095                                          operation
1491                version of github enterprise server
1534                        attacker with low privilege
628                        control of affect system.the
1379                             specially craft packet
614                         specially craft application
319                                specially craft file
238     aka window elevation of privilege vulnerability
594                                orchestrator service
590                                     user experience
709                                                fail
```

[132]:

```
final_all=pd.
 ↪concat([final_cwe_impact_note,final_cwe_dis,final_cwe_ex_dis,final_CVE_des],axis=0).
 ↪drop_duplicates()
```

```
[134]: final_cwe_impact_note.to_csv('final_cwe_impact_note.csv',index=False)
```

```
[135]: final_cwe_dis.to_csv('final_cwe_dis.csv',index=False)
```

```
[136]: final_all.to_csv('final_all.csv',index=False)
```

load those csv to google drive

```
[ ]:
```

1.3 make triples -Method 2 general

```
[46]: from spacy.matcher import Matcher
def get_entities(sent):
    ## chunk 1
    ent1 = ""
    ent2 = ""
    prv_tok_dep = ""  # dependency tag of previous token in the sentence
    prv_tok_text = ""  # previous token in the sentence
    prefix = ""
    modifier = ""
    #############################################################
    for tok in nlp_m(sent):
        ## chunk 2
        # if token is a punctuation mark then move on to the next token
        if tok.dep_ != "punct":
            # check: token is a compound word or not
            if tok.dep_ == "compound":
                prefix = tok.text
                # if the previous word was also a 'compound' then add the␣
 ↪current word to it
                if prv_tok_dep == "compound":
                    prefix = prv_tok_text + " " + tok.text
            # check: token is a modifier or not
            if tok.dep_.endswith("mod") == True:
                modifier = tok.text
                # if the previous word was also a 'compound' then add the␣
 ↪current word to it
                if prv_tok_dep == "compound":
                    modifier = prv_tok_text + " " + tok.text
            ## chunk 3
            if tok.dep_.find("subj") == True:
                ent1 = modifier + " " + prefix + " " + tok.text
                prefix = ""
                modifier = ""
                prv_tok_dep = ""
                prv_tok_text = ""
                ## chunk 4
            if tok.dep_.find("obj") == True:
                ent2 = modifier + " " + prefix + " " + tok.text
            ## chunk 5
            # update variables
            prv_tok_dep = tok.dep_
            prv_tok_text = tok.text
        else:
            break
    #############################################################
    return [ent1.strip(), ent2.strip()]

def get_relation(sent):
    # nlp = spacy.load('en_core_web_sm')
    doc = nlp_m(sent)
    # Matcher class object
    matcher = Matcher(nlp.vocab)
    #define the pattern
    pattern = [{'DEP':'ROOT'},
            {'DEP':'prep','OP':"?"},
            {'DEP':'agent','OP':"?"},
            {'POS':'ADJ','OP':"?"}]
    matcher.add("matching_1", None, pattern)
    matches = matcher(doc)
    k = len(matches) - 1
    span = doc[matches[k][1]:matches[k][2]]
    return(span.text)


def process_text(text):

    doc = nlp_m(text)

    sentences = [sent.string.strip() for sent in doc.sents]
    relations = [get_relation(i) for i in sentences]

    entity_pairs = []

    for i in sentences:
        entity_pairs.append(get_entities(i))

    source = [i[0] for i in entity_pairs]
    target = [i[1] for i in entity_pairs]
    kg_df = pd.DataFrame({'source':source, 'target':target, 'edge':relations})
    output = zip(source, relations, target)
    return(list(output))
```

[53]:

```
x=' '.join(list(cwe_impact_note['str'].drop_duplicates())).lower().
 ↪replace('\n','').replace('\t','')
```

[54]:

```
method2_triple=pd.DataFrame(process_text(x),columns=['e1','r','e2'])
```

[55]:

```
method2_triple[(method2_triple.e1 != '') & (method2_triple.r != '') &␣
 ↪(method2_triple.e2 != '')  ]
```

[55]:

```
                    e1         r                              e2
1               cookie     allow                        question
2             attacker  redirect               malicious  website
4           first  step        be                     such  steal
5             attacker     mimic                  malicious  site
7       perform  that     trick                             view
..                 …         …                               …
680           attacker   exploit                          cookie
681  damaging  attack   include                    end user file
683           attacker   be able                remote  location
685           attacker      gain                sensitive  datum
686             access     allow  sensitive configuration information

[494 rows x 3 columns]
```

[ ]:

```

```

1.4 make triples -Method 3 Noun Chunks and Verb Span

[66]:

```
from spacy.tokens import Span
```

[81]:

```
nlp_m = spacy.load("en_core_web_sm")
```

[60]:

```
#Verb span

matcher = Matcher(nlp_m.vocab)

#define the pattern
pattern = [{'DEP':'ROOT'},
           {'DEP':{'REGEX': 'aux|prep'},'OP':"?"},
#         {'DEP':'prep','OP':"?"},
#         {'DEP':'agent','OP':"?"},
     {'POS':'ADJ','OP':"?"},
#         {'DEP':'attr','OP':"?"},
        {'DEP':'aux','OP':"?"},
#         {'DEP':'xcomp','OP':"?"},
          {'DEP':'xcomp','OP':"?"}
          ]

matcher.add("matching_1", None, pattern)
# matches = matcher(doc)
```

[82]:

```
with doc.retokenize() as retokenizer:
    #match and tag volume units
    matches = matcher(doc)
    for match_id, start, end in matches:
        span = Span(doc, start, end)

        try:
            if len(span) > 1:
                retokenizer.merge(span)
        except ValueError:
            pass
```

[67]:

```
# noun chunk
merge_nps = nlp_m.create_pipe("merge_noun_chunks")
nlp_m.add_pipe(merge_nps)
```

[ ]:

```

```

[74]:

```
def get_entities(sent):
    ## chunk 1
    ent1 = ""
    ent2 = ""
    relation=""

    #############################################################
    for tok in nlp_m(sent):

        if tok.dep_ != "punct":

            if tok.dep_.find("subj") == True:
                ent1 =  tok.text

            if tok.dep_.find("obj") == True:
                ent2 =  tok.text



            if tok.dep_=="ROOT":
                relation=tok.text
            if ent1 !='' and ent2 !='' and relation != '':
                break
    #############################################################
    return [ent1.strip(), ent2.strip(),relation.strip()]


def process_text(text):

    doc = nlp_m(text)

    sentences = [sent.string.strip() for sent in doc.sents]
    relations = [get_relation(i) for i in sentences]

    entity_pairs = []

    for i in sentences:
        entity_pairs.append(get_entities(i))

    source = [i[0] for i in entity_pairs]
    target = [i[1] for i in entity_pairs]
    kg_df = pd.DataFrame({'source':source, 'target':target, 'edge':relations})
    output = zip(source, relations, target)
    return(list(output))
```

[76]:

```
m3=pd.DataFrame(process_text(x),columns=['e1','r','e2'])
```

[79]:

```
m3=m3[(-m3.r.isin(['is possible','be able','be in','is','be'])) &(m3.e2 !='') ]
```

[80]:

```
m3
```

[80]:

```
                              e1            r                       e2
0    sensitive information store    expose to               the cookie
1                            set        allow        the httponly flag
2                    an attacker     redirect                   a user
5                   the attacker        mimic         a malicious site
6                    an attacker       create               a username
..                           …          …                      …
681        other damaging attack      include           the disclosure
682              the inject code       access    restrict datum / file
684                           it  be possible  normal program behavior
685                  an attacker         gain                   access
686                       access        allow                     file

[506 rows x 3 columns]
```

[ ]:

```

```

[ ]:

```

```

2 neo4j graph

[137]:

```
from py2neo import Graph
```

[145]:

```
graph = Graph('bolt://54.237.11.242:
 ↪32819',user='neo4j',password='appeals-energy-shoulder')
```

[ ]:

```
# load csv from google drive
```

[146]:

```
# CWE impact

graph.delete_all()
query="""CREATE CONSTRAINT ON (c1:node) ASSERT c1.id IS UNIQUE"""
graph.run(query)

query="""
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'https://docs.google.com/spreadsheets/d/e/
 ↪2PACX-1vRGkhYjxQ8OW-5miKxPzVuuZOX4bGcN2zFtZS7KLD8YawlenoML0VLa9xTtJOnvGjvSMHo9-bfVAeiM/
 ↪pub?gid=1613950039&single=true&output=csv' AS row
MERGE(c1:node{id:row.e1})"""
graph.run(query)

query="""
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'https://docs.google.com/spreadsheets/d/e/
 ↪2PACX-1vRGkhYjxQ8OW-5miKxPzVuuZOX4bGcN2zFtZS7KLD8YawlenoML0VLa9xTtJOnvGjvSMHo9-bfVAeiM/
 ↪pub?gid=1613950039&single=true&output=csv' AS row
MERGE(c1:node{id:row.e2})"""
graph.run(query)

query="""
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'https://docs.google.com/spreadsheets/d/e/
 ↪2PACX-1vRGkhYjxQ8OW-5miKxPzVuuZOX4bGcN2zFtZS7KLD8YawlenoML0VLa9xTtJOnvGjvSMHo9-bfVAeiM/
 ↪pub?gid=1613950039&single=true&output=csv' AS row
MATCH(c1:node{id:row.e1})
MATCH(c2:node{id:row.e2})
CREATE unique (c1)-[:relation{id:row.r}]->(c2)
"""
graph.run(query)
```

[146]:

```
(No data)
```

[147]:

```
# CWE dis

graph.delete_all()
```

```
query="""CREATE CONSTRAINT ON (c1:node) ASSERT c1.id IS UNIQUE"""
graph.run(query)

query="""
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'https://docs.google.com/spreadsheets/d/e/
 ↪2PACX-1vRC3fMmCXuQczkQZSwysZvOT6o4mjk1ho3J_xMQiIMUYYSGgD1zjs7UxGlEKrEwa6WlvRErBD85V-Gu/
 ↪pub?gid=134573794&single=true&output=csv' AS row
MERGE(c1:node{id:row.e1})"""
graph.run(query)

query="""
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'https://docs.google.com/spreadsheets/d/e/
 ↪2PACX-1vRC3fMmCXuQczkQZSwysZvOT6o4mjk1ho3J_xMQiIMUYYSGgD1zjs7UxGlEKrEwa6WlvRErBD85V-Gu/
 ↪pub?gid=134573794&single=true&output=csv' AS row
MERGE(c1:node{id:row.e2})"""
graph.run(query)

query="""
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'https://docs.google.com/spreadsheets/d/e/
 ↪2PACX-1vRC3fMmCXuQczkQZSwysZvOT6o4mjk1ho3J_xMQiIMUYYSGgD1zjs7UxGlEKrEwa6WlvRErBD85V-Gu/
 ↪pub?gid=134573794&single=true&output=csv' AS row
MATCH(c1:node{id:row.e1})
MATCH(c2:node{id:row.e2})
CREATE unique (c1)-[:relation{id:row.r}]->(c2)
"""
graph.run(query)
```

```
[147]: (No data)
```

```
[ ]: # all

graph.delete_all()
query="""CREATE CONSTRAINT ON (c1:node) ASSERT c1.id IS UNIQUE"""
graph.run(query)

query="""
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'https://docs.google.com/spreadsheets/d/e/
 ↪2PACX-1vQ_wYfrtavNHFTYHCjF-UuYsoFjeqadWcMpDyhF9iaavZuJwwEpwkR-XLca24bkAjFczp1L35_ArP6t/
 ↪pub?gid=910519178&single=true&output=csv' AS row
MERGE(c1:node{id:row.e1})"""
graph.run(query)

query="""
```

```
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'https://docs.google.com/spreadsheets/d/e/
 ↪2PACX-1vQ_wYfrtavNHFTYHCjF-UuYsoFjeqadWcMpDyhF9iaavZuJwwEpwkR-XLca24bkAjFczp1L35_ArP6t/
 ↪pub?gid=910519178&single=true&output=csv' AS row
MERGE(c1:node{id:row.e2})"""
graph.run(query)

query="""
USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'https://docs.google.com/spreadsheets/d/e/
 ↪2PACX-1vQ_wYfrtavNHFTYHCjF-UuYsoFjeqadWcMpDyhF9iaavZuJwwEpwkR-XLca24bkAjFczp1L35_ArP6t/
 ↪pub?gid=910519178&single=true&output=csv' AS row
MATCH(c1:node{id:row.e1})
MATCH(c2:node{id:row.e2})
CREATE unique (c1)-[:relation{id:row.r}]->(c2)
"""
graph.run(query)
```

2.1 query for graph

[ ]:

```
# graph for cwe-impact: attacker
# MATCH (p)-[r]->(m) where p.id='attacker' RETURN p,r,m LIMIT 30

# all data graph query
# MATCH (p)-[r*3..6]->(m) RETURN p,r,m LIMIT 125
```

[ ]:

```

```