pacu / pacu / modules / guardduty__whitelist_ip / main.py

Ryan Gerstenkorn Support for packaging (#247) 743f9c9 · 3 years ago

101 lines (88 loc) · 5.31 KB

Code Blame Raw

```
#!/usr/bin/env python3
import argparse
from botocore.exceptions import ClientError
import copy
import string
import random


module_info = {
    'name': 'guardduty__whitelist_ip',
    'author': 'Spencer Gietzen',
    'category': 'EVADE',
    'one_liner': 'Adds an IP address to the list of trusted IPs in GuardDuty.',
    'description': 'This module accepts a file containing IPv4 addresses and adds them to the Guarc
    'services': ['GuardDuty'],
    'prerequisite_modules': ['detection__enum_services'],
    'external_dependencies': [],
    'arguments_to_autocomplete': ['--path', '--regions', '--targets'],
}

parser = argparse.ArgumentParser(add_help=False, description=module_info['description'])
parser.add_argument('--path', required=True, help='A public link to a file containing a list of IP
parser.add_argument('--regions', required=False, default=None, help='The set of regions to target G
parser.add_argument('--targets', required=False, default=None, help='Comma-separated list of GuardD


```

```python
def main(args, pacu_main):
    session = pacu_main.get_active_session()
    args = parser.parse_args(args)
    print = pacu_main.print
    input = pacu_main.input
    fetch_data = pacu_main.fetch_data
    get_regions = pacu_main.get_regions

    data = {'detectors': [], 'ip_sets': []}

    if args.targets:
        detectors = []
        regions = []
        targets = args.targets.split(',')
        for target in targets:
            id, region = target.split('@')
            detectors.append({'Id': id, 'Region': region})
            regions.append(region)
        regions = list(set(regions))
    else:
        regions = get_regions('GuardDuty')
        if fetch_data(['GuardDuty', 'Detectors'], module_info['prerequisite_modules'][0], '--guard-
            print('Pre-req module failed.')
            return
        detectors = copy.deepcopy(session.GuardDuty['Detectors'])

    for region in regions:
        client = pacu_main.get_boto3_client('guardduty', region)
        for detector in detectors:
            if detector['Region'] == region:
                print(' ({}) Detector {}:'.format(region, detector['Id']))
                data['detectors'].append(detector)
                try:
                    response = client.create_ip_set(
                        Activate=True,
                        DetectorId=detector['Id'],
                        Format='TXT',
                        Location=args.path,
                        Name=''.join(random.choice(string.ascii_lowercase + string.digits) for _ ir
                    )
                    ip_set_id = response['IpSetId']
                    data['ip_sets'].append(ip_set_id)
                    print('    Created IPSet: {}'.format(ip_set_id))
                except ClientError as error:
                    if 'an attempt to create resources beyond the current AWS account limits' in st
                        print('    Error: Existing IPSet found')
```

```
73                              print('    WARNING: Replacing an existing IPSet could have unintended bad c
74                              remove = input('Try to replace the IPSet? (y/n) ')
75                              if remove.strip() == 'y':
76                                  try:
77                                      response = client.list_ip_sets(
78                                          DetectorId=detector['Id']
79                                      )
80                                      # There is a max of one IPSet per detector
81                                      existing_ip_set_id = response['IpSetIds'][0]
82
83                                      client.update_ip_set(
84                                          Activate=True,
85                                          DetectorId=detector['Id'],
86                                          Location=args.path,
87                                          IpSetId=existing_ip_set_id
88                                      )
89
90                                      print('      Replaced IPSet {}...\n'.format(existing_ip_set_id))
91                                      data['ip_sets'].append(existing_ip_set_id)
92                                  except ClientError as error:
93                                      print('      Error: {}'.format(str(error)))
94                          else:
95                              print('      Error: {}'.format(str(error)))
96
97          return data
98
99
100     def summary(data, pacu_main):
101         return '{} IPSet(s) created for {} GuardDuty Detector(s).'.format(len(data['ip_sets']), len(dat
```