research/check.py at main · trendmicro/research · GitHub

github.com/trendmicro/research/blob/main/cyclops_blink/c2-scripts/check.py

trendmicro

trendmicro/
research

1 Contributor 0 Issues 1 Star 0 Forks

```
#!/usr/bin/env python3

import socket
import ssl
import sys
import requests
from pathlib import Path

def usage():
print("Usage:\n\t{0} HOST[:]PORT\n\nExamples:\n\t{0} 8.8.8.8 443\n\t{0} 9.9.9.9:666\n".format(Path(__file__).name))
sys.exit(1)

def myip():
r = requests.get('https://api.ipify.org?format=json')
```

```
    return r.json()['ip']

def check_cyclops_blink_c2(hostname, port, extaddr):
    ctx = ssl.create_default_context()
    ctx.check_hostname = False # Disables hostname checking
    ctx.verify_mode = ssl.CERT_NONE # Do not verify the certificate

    veredict = 'NOT DETECTED'
    response = ''

    try:
        with socket.create_connection((hostname, port), timeout=5) as sock:
            with ctx.wrap_socket(sock, server_hostname=hostname) as ssock:
                ssock.settimeout(10)
                ssock.send(b'\x00\x00\x00\x08')
                response = ssock.read(2048)
                if len(response) == 4:
                    veredict = 'POSSIBLE'
                    if socket.inet_ntoa(response) == extaddr:
                        veredict = 'ACTIVE'
                ssock.close()
    except:
        veredict = 'UNREACHABLE'

    print(hostname,
          port,
          len(response),
          response,
          veredict)
```

```
def main(argv):
if len(argv) < 2:
usage()

# Accepts both host:port or host<space>port
pos = sys.argv[1].find(':')
if pos != -1:
hostname = sys.argv[1][:pos]
port = sys.argv[1][pos+1:]
else:
if len(argv) < 3:
usage()
hostname = sys.argv[1]
port = sys.argv[2]

check_cyclops_blink_c2(hostname, port, myip())

if __name__ == "__main__":
main(sys.argv)
```