

nicolargo / glances Public

<> Code Issues 90 Pull requests 12 Discussions Actions Projects

Commit d6808be

nicolargo committed last week · ✖ 17 / 19

SSRF in Glances IP Plugin via public_api leads to credential leakage - Correct CVE-2026-35587

develop · v4.5.4

1 parent [fdfb977](#) commit d6808be

1 file changed +17 -1 lines changed

↑ Top ⚙

Filter files...

glances/plugins/ip

__init__.py

1 file changed +17 -1 lines changed

Search within code ⚙

glances/plugins/ip/__init__.py

@@ -9,6 +9,8 @@

9 9 """IP plugin."""

10 10

11 11 import threading

12 + from urllib.parse import urlparse

13 + from urllib.request import Request, urlopen

12 14

13 15 from glances.globals import get_ip_address, json_loads, urlopen_auth

14 16 from glances.logger import logger

@@ -71,6 +73,17 @@ def __init__(self, args=None, config=None):

71 73 or self.public_api is None

72 74 or self.public_field is None

```

73     75         )
76     +
77     +         # Defence-in-depth: validate URL scheme to prevent SSRF via config
78     +         if not self.public_disabled and self.public_api:
79     +             parsed = urlparse(self.public_api)
80     +             if parsed.scheme not in ('http', 'https'):
81     +                 logger.warning(
82     +                     f"IP plugin - public_api uses forbidden scheme
83     +                     '{parsed.scheme}://', "
84     +                     "only http:// and https:// are allowed. Public IP
85     +                     disabled."
86     +                 )
87     +                 self.public_disabled = True
88
89     +         self.public_address_refresh_interval = self.get_conf_value(
90     +             "public_refresh_interval",
91     +             default=self._default_public_refresh_interval
92     +         )
93
94     +     @@ -258,7 +271,10 @@ def run(self):
95
96     +     def _fetch_public_ip_info(self):
97     +         """Fetch public IP information from the configured API."""
98     +         try:
99     +             response = urlopen_auth(self.url, self.username, self.password,
100     +                 self.timeout).read()
101     +             if self.username and self.password:
102     +                 response = urlopen_auth(self.url, self.username, self.password,
103     +                     self.timeout).read()
104     +             else:
105     +                 response = urlopen(Request(self.url),
106     +                     timeout=self.timeout).read()
107     +             return json_loads(response)
108     +         except Exception as e:
109     +             logger.debug(f"IP plugin - Cannot get public IP information from
110     +                 {self.url} ({e})")
111
112     +

```

Comments 0



Please [sign in](#) to comment.