VULNSPACE

EXTERNAL ATTACK SURFACE MANAGEMENT

EASM
BY LOGICS 7

Prevents cyberattacks that can lead to financial and data loss

Discovers the attack surface, including servers, mobile, web apps, cloud, container, IoT and OT assets

Automates daily vulnerability scans for all targets

Includes expert service to remove false-positives, check for 1-days and triage the vulnerabilities

Boosts your cybersecurity team with weekly meetings conducted by our pentesters

EASM
BY LOGICS 7

When do you need a vulnerability scan

- Before publishing a new IT-system
- As part of a network management process
- Regulary to be industry compliant
- As a part of a software development cycle
- To protect against 1-day exploits as Log4j, BadRabbit and Proxy Logon

EASM
BY LOGICS 7

Checks

- Check all TCP and UDP ports to be compliant to allowlist
- Scan services with the database of 110,807 vulnerabilities and exploits from MITER, NIST and ExploitDB
- Bruteforce passwords for 40+ protocols
- Run next-generation web crawler for modern web-apps
- Test for OWASP-TOP-10 web vulnerabilities
- Find vulnerabilities for frontend components
- Run API tests based on the OpenAPI \swagger specification

EASM
BY LOGICS 7

L3-L7 scans

FULL COVERAGE

Integrations

- Discover organization's IPs, domains, and subnets from DNS, PTR, RIPE, search engines, and crt.sh
- Import the targets from an asset management system
- Use HTTP REST API to schedule, run, and gather reports
- Export scan results in JSON and CSV
- Import allowed ports list for hosts and networks
- Use your internal storage to store sensitive scan results
- Run custom scans with our Python wrapper

EASM
BY LOGICS 7

```python
class Scanner(object) :
    name = "scanner_base_object"
    vuln_type = "default_vuln_type"

    def __init__ (self, opts, target, metadata, vulnerability_body_fields_to_web_interface)
        self.Vulnerability_body_fields_to_web_interface = vulnerability_body_fields_to_web_interface
        self.opts = opts
        self.target = target
        self.metadata = metadata
        self.circuit(self. target)

    @staticmethod
    def circuit (target):
        """..."""
        return Vulnerability ()

    def check_start_condition (self):
        """..."""
        return True

    class ScannerError(Exception):
        def __init__ (self, value):
            self.value = value
        def __str__ (self):
            return repr(self.value)
```

EASM BY LOGICS 7

Python wrapper

WRAP YOUR OWN SECURITY TOOLS TO RUN IT IN OUR CLOUD

Get 14 days free trial at
logics7.com/easm