r3naissance Update http-vuln-cve2020-13968.nse History

1 contributor

109 lines (97 sloc) | 3.65 KB

```
local http = require "http"
local shortport = require "shortport"
local string = require "string"
local stdnse = require "stdnse"
local vulns = require "vulns"
local table = require "table"

description = [[
CRK Business Platform - CVE-2020-13968 - SQL Injection on versions <= 2019.1
]]

---
-- @usage nmap --script http-vuln-cve2020-13968 -p 443 <target>
-- @output
-- PORT   STATE SERVICE VERSION
-- 443/tcp open  http
-- | http-vuln-cve2020-13968:
-- |   VULNERABLE:
-- |   CRK Business Platform - SQL Injection on versions <= 2019.1
-- |       State: VULNERABLE
-- |     IDs:  CVE:CVE-2020-13968
-- |     Risk factor: High
-- |       Unauthenticated users can inject SQL statements against the DB
-- |       on any path using the 'strSessao' parameter.
-- |
-- |     Disclosure date: 2020-06-08
-- |     References:
-- |_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-13968
--
--
-- @xmloutput
-- <table key="CVE-2020-13968">
-- <elem key="title">CRK Business Platform - SQL Injection on versions <= 2019.1</elem>
-- <elem key="state">VULNERABLE</elem>
-- <table key="ids">
-- <elem>CVE:CVE-2020-13968</elem>
-- </table>
-- <table key="description">
-- <elem>Unauthenticated user can inject SQL statements against the DB on any path using the 'strSessao' parameter.</elem>
-- </table>
-- <table key="dates">
-- <table key="disclosure">
-- <elem key="day">08</elem>
-- <elem key="month">06</elem>
-- <elem key="year">2020</elem>
-- </table>
-- </table>
-- <elem key="disclosure">2020-06-08</elem>
-- <table key="check_results">
-- </table>
-- <table key="extra_info">
-- </table>
-- <table key="refs">
-- <elem>https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-13968</elem>
-- </table>
-- </table>
--
---

author = "Chapman (R3naissance) Schleiss"
license = "Same as Nmap--See https://nmap.org/book/man-legal.html"
categories = {"vuln", "intrusive"}

-- aquatone xlarge ports
portrule = shortport.port_or_service( {80, 81, 300, 443, 591, 593, 832, 981, 1010, 1311, 2082, 2087, 2095, 2096, 2480, 3000, 3128, 3333, 4243, 4567, 4711, 4712, 4993, 5000, 5104,

action = function(host, port)
  local function inject(payload)
    options = {}
    options['timeout'] = 1000
    local uri = vuln_uri .. payload

    local response = http.get(host, port, uri, options)
    stdnse.debug1("Response %s", response.status)

    if string.match(response.body, "Invalid object name") then
      message = "Single quote SQL statement breakout found"
      stdnse.debug1(response.body)
```

```
    end
  end

  local vuln_table = {
    title = "CRK Business Platform - SQL Injection",
    IDS = {CVE = 'CVE-2020-13968'},
    risk_factor = "High",
    references = {
        'https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-13968'
    },
    dates = {
      disclosure = {year = '2020', month = '06', day = '08'},
    },
    check_results = {},
    extra_info = {}
  }

  local vuln_report = vulns.Report:new(SCRIPT_NAME, host, port)
  vuln_table.state = vulns.STATE.NOT_VULN
  vuln_uri = stdnse.get_script_args(SCRIPT_NAME..".uri") or '/sistemas/administrativo/CRK.GerenciadorAcesso/operacoes/alterar_senha/alterar_senha.aspx?Mod=P&Idioma=pt-br&IDFuncao=2

  if pcall(inject, "'") then
    vuln_table.state = vulns.STATE.VULN
    table.insert(vuln_table.extra_info, message)
  else
    stdnse.debug1("Could not find error in response")
  end

  return vuln_report:make_output(vuln_table)

end
```