The RSA PKCS1 v1.5 decryption implementation does not detect ciphertext modification (prepended 0's bytes to the ciphertext) #439

Closed **adelapie** opened this issue on Jun 6, 2020 · 3 comments

Labels bug

adelapie commented on Jun 6, 2020

The jsrsasign 8.0.16 RSA PKCS1 v1.5 decryption implementation does not detect prepended 0's bytes to the ciphertext and accepts modified ciphertexts without error.

You can verify this using the following test vectors from Google Wycheproof:

```
{
  "algorithm" : "RSAES-PKCS1-v1_5",
  "generatorVersion" : "0.8r12",
  "numberOfTests" : 65,
  "header" : [
    "Test vectors of type RsaesPkcs1Decrypt are intended to check the decryption",
    "of RSA encrypted ciphertexts."
  ],
  "notes" : {
    "InvalidPkcs1Padding" : "This is a test vector with an invalid PKCS #1 padding. Implementations must ensure that different error conditions cannot be distinguished, since the
information about the error condition can be used for a padding oracle attack. (RFC 8017 Section 7.2.2)"
  },
  "schema" : "rsaes_pkcs1_decrypt_schema.json",
  "testGroups" : [
    {
      "d" :
"1a502d0eea6c7b69e21d5839101f705456ed0ef852fb47fe21071f54c5f33c8ceb066c62d727e32d26c58137329f89d3195325b795264c195d85472f7507dbd0961d2951f935a26b34f0ac24d15490e1128a9b7138915bc7dbfa8f
      "e" : "010001",
      "keysize" : 2048,
      "n" :
"00b3510a2bcd4ce644c5b594ae5059e12b2f054b658d5da5959a2fdf1871b808bc3df3e628d2792e51aad5c124b43bda453dca5cde4bcf28e7bd4effba0cb4b742bbb6d5a013cb63d1aa3a89e02627ef5398b52c0cfd97d208abeb
      "privateKeyJwk" : {
        "alg" : "RSA1_5",
        "d" : "GlAtDupse2niHVg5EB9wVFbtDvhS-0f-IQcfVMXzPIzrBmxi1yfjLSbFgTcyn4nTGVMlt5UmTBldhUcvdQfb0JYdKVH5NaJrNPCsJNFUkOESiptxOJFbx9v6j-OWNXExxUOunJhQc2jZzrCMHGGYo-
2nrqGFoOl2zULCLQDwA9nxnZbqTJr8v-
FEHMyALPsGifWdgExqTk9ATBUXR0XtbLi8iO8LM7oNKoDjXkO8kPNQBS5yAW51sA01ejgcnA1GcGnKZgiHyYd2Y0n8xDRgtKpRa84Hnt2HuhZDB7dSwnftlSitO6C_GHc0ntO3lmpsJAEQQJv00PreDGj9rdhH_Q",
        "dp" : "lql5jSUCY0ALtidzQogWJ-B87N-RGHsBuJ_0cxQYinwg-ySAAVbSyF1WZujfbO_5-YBN362A_1dn3lbswCnHK_bHF9-fZNqvwprPnceQj5oK1n4g6JSZNsy6GNAhosT-uwQ0misgR8SQE4W25dDGkdEYsz-
BgCsyrCcu8J5C-tU",
        "dq" : "BVT0GwuH9opFcis74M9KseFlA0wakQAquPKenvni2rb-57JFW6-
0IDfp0vflM_NIoUdBL9cggL58JjP12ALJHDnmvOzj5nXlmZUDPFVzcCDa2eizDQS4KK37kwStVKEaNaT1BwmHasWxGCNrp2pNfJopHdlgexad4dGCOFaRmZ8",
        "e" : "AQAB",
        "kid" : "none",
        "kty" : "RSA",
        "n" : "s1EKK81M5kTFtZSuUFnhKy8FS2WNXaWVmi_fGHG4CLw98-Yo0nkuUarVwSS0O9pFPcpc3kvPKOe9Tv-6DLS3Qru21aATy2PRqjqJ4CYn71OYtSwM_ZfSCKvrjXybzgu-sBmobdtYm-sppbdL-GEHXGd8gdQw8DDCZSR6-
dPJFAzLZTCdB-Ctwe_RXPF-ewVdfaOGjkZIzDoYDw7n-OHnsYCYozkbTOcWHpjVevipR-IBpGPi1rvKgFnlcG6d_tj0hWRl_6cS7RqhjoiNEtxqoJzpXs_Kg8xbCxXbCchkf11STA8udiCjQWuWI8rcDwl69XMmHJjIQAqhKvOOQ8rYTQ",
        "p" : "7BJc834xCi_0YmO5suBinWOQAF7IiRPU-3G9TdhWEkSYquupg9e6K9lC5k0iP-t6I69NYF7-6mvXDTmv6Z01o6oV50oXaHeAk74O3UqNCbLe9tybZ_-FdkYlwuGSNttMQBzjCiVy0-y0-
Wm3rRnFIsAtd0RlZ24aN3bFTWJINIs",
        "q" : "wnQqvNmJe9SwtnH5c_yCqPhKv1cF_4jdQZSGI6_p3KYNxlQzkHZ_6uvrU5V27ov6YbX8vKlKfO91oJFQxUD6lpTdgAStI3GMiJBJIZNpyZ9EWNSvwUj28H34cySpbZz3s4XdhiJBShgy-
fKURvBQwtWmQHZJ3EGrcOI7PcwiyYc",
        "qi" : "HGQBidm_6MYjgzIQp2xCDG9E5ddg4lmRbOwq4rFWRWlg_ZXidHZgw4lWIlDwVQSc-
rflwwOVSThKeiquscgk069wlIKoz5tYcCKgCx8HIttQ8zyybcIN0iRdUmXfYe4pg8k4whZ9zuEh_EtEecI35yjPYzq2CowOzQT85-O6pVk"
      },
      "privateKeyPem" : "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIBAAKCAQEAs1EKK81M5kTFtZSuUFnhKy8FS2WNXaWVmi/fGHG4CLw98+Yo\n0nkuUarVwSS0O9pFPcpc3kvPKOe9Tv+6DLS3Qru21aATy2PRqjqJ4CYn71OYtSwM\n/ZfSCKvrjXybzgu+sBmobdtYm+sppbdL+GEHXGd8gdQw8DDCZ
----END RSA PRIVATE KEY-----",
      "privateKeyPkcs8" :
"308204bd020100300d06092a864886f70d0101010500048204a7308204a30201000282010100b3510a2bcd4ce644c5b594ae5059e12b2f054b658d5da5959a2fdf1871b808bc3df3e628d2792e51aad5c124b43bda453dca5cde4b
      "type" : "RsaesPkcs1Decrypt",
      "tests" : [
        {
          "tcId" : 31,
          "comment" : "prepended bytes to ciphertext",
          "msg" : "54657374",
          "ct" :
"00004501b4d669e01b9ef2dc800aa1b06d49196f5a09fe8fbcd037323c60eaf027bfb98432be4e4a26c567ffec718bcbea977dd26812fa071c33808b4d5ebb742d9879806094b6fbeea63d25ea3141733b60e31c6912106e1b758a
          "result" : "invalid",
          "flags" : []
        },
         {
          "tcId" : 31,
          "comment" : "correct ciphertext",
          "msg" : "54657374",
          "ct" :
"4501b4d669e01b9ef2dc800aa1b06d49196f5a09fe8fbcd037323c60eaf027bfb98432be4e4a26c567ffec718bcbea977dd26812fa071c33808b4d5ebb742d9879806094b6fbeea63d25ea3141733b60e31c6912106e1b758a7fe0
          "result" : "valid",
          "flags" : []
        }
      ]
    }
  ]
}
```

and proof of concept:

```
var rs = require('jsrsasign');
var obj = require("./rsa_pkcs1.json");
```

```
for (let testGroup of obj.testGroups) {

    var keyPem = testGroup.privateKeyPem;

    var prv = new rs.RSAKey();
    prv.readPrivateKeyFromPEMString(keyPem);

    for(let test of testGroup.tests) {
     console.log("[*] Test " + test.tcId + " result: " + test.result)

     try {

      var pt = rs.crypto.Cipher.decrypt(test.ct, prv);
      var result = Buffer.from(pt).toString('hex') === test.msg;

     if (result == true) {
      if (test.result == "valid" || test.result == "acceptable")
       console.log("Result: PASS");
      else
       console.log("Result: FAIL")
     }

     if (result == false) {
      if (test.result == "valid" || test.result == "acceptable")
       console.log("Result: FAIL");
      else
       console.log("Result: PASS")
     }

     } catch (e) {
      console.log("ERROR - VERIFY: " + e)

      if (test.result == "valid" || test.result == "acceptable")
       console.log("Result: FAIL");
      else
       console.log("Result: PASS")

     }

    }
}
```

with result:

```
[*] Test 31 result: invalid
Result: FAIL
[*] Test 31 result: valid
Result: PASS
```

Best regards,
Antonio

kjur commented on Jun 20, 2020 Owner

Thank you for your report. This issue was fixed in the 8.0.18 release today.

kjur closed this as completed on Jun 20, 2020

adelapie commented on Jun 22, 2020 Author

CVE-2020-14967 is assigned to this issue with the following description: An issue was discovered in the jsrsasign package before 8.0.18 for Node.js. Its RSA PKCS1 v1.5 decryption implementation does not detect ciphertext modification by prepending '\0' bytes to ciphertexts (it decrypts modified ciphertexts without error). An attacker might prepend these bytes with the goal of triggering memory corruption issues.

kjur commented on Jun 23, 2020 Owner

jsrsasign security advisory (2020-Jun-24):
CVE-2020-14967
RSA RSAES-PKCS1-v1_5 and RSA-OAEP decryption vulnerability with prepending zeros
GHSA-xxxq-chmp-67g4

kjur added the bug label on Aug 18, 2020

This was referenced on Mar 13, 2021

Bump jsrsasign from 8.0.12 to 8.0.19 m0rphtail/Teleport#6
Closed

Bump jsrsasign from 8.0.12 to 8.0.19 Cyper77/CyberChef#1
Closed

Assignees

No one assigned

Labels

bug

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants