Talos Vulnerability Report

TALOS-2020-1189

Genivia gSOAP WS-Security plugin denial-of-service vulnerability

JANUARY 5, 2021

CVE NUMBER

CVE-2020-13578

Summary

A denial-of-service vulnerability exists in the WS-Security plugin functionality of Genivia gSOAP 2.8.107. A specially crafted SOAP request can lead to denial of service. An attacker can send an HTTP request to trigger this vulnerability.

Tested Versions

Genivia gSOAP 2.8.107

Product URLs

https://www.genivia.com/products.html#gsoap

CVSSv3 Score

7.5 - CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H

CWE

CWE-476 - NULL Pointer Dereference

Details

The gSOAP toolkit is a C/C++ library for developing XML-based web services. It includes several plugins to support the implementation of SOAP and web service standards. The framework also provides multiple deployment options including modules for both IIS and Apache, standalone CGI scripts and its own standalone HTTP service.

One of the many plugins provided by gSOAP includes the wsse plugin for supporting the WS-Security specification. The BinarySecurityToken element can be used to provide encoded X509 certficates. The BinarySecurityToken element includes both a EncodingType and a ValueType attribute which can include URIs that defined the data format for this field. Due to an uninitialized pointer being used during the processing of this element, it is possible to trigger a denial of service condition depending on the pre-existing data that the location of this pointer at the time of creation.

The pointer is first created within soap_wsse_get_BinarySecurityTokenX509.

```
soap_wsse_get_BinarySecurityTokenX509(struct soap *soap, const char *id)
{
  X509 *cert = NULL;
  char *valueType = NULL;
#if (OPENSSL_VERSION_NUMBER >= 0x0090800fL)
  const unsigned char *data;  <------- Not initialized
#else
  unsigned char *data;  <------- Not initialized
#endif
  int size;
  DBGFUN1("soap_wsse_get_BinarySecurityTokenX509", "id=%s", id?id:"");
  if (!soap_wsse_get_BinarySecurityToken(soap, id, &valueType, (unsigned char**)&data, &size) <----- data should be initialized during this
call
   && valueType
   && !strcmp(valueType, wsse_X509v3URI))
    cert = d2i_X509(NULL, &data, size); <------ Depending on the data that data points to, a number of access violations can occur here.
  /* verify the certificate */
  if (cert && soap_wsse_verify_X509(soap, cert))
  {
    X509_free(cert);
    cert = NULL;
  }
  return cert;
}

The parser attempts to decode the token based on the EncodingType.  If Encoding type is set but doesn't match one of the two hardcoded URIs,
data will also not be set here.
Finally, because the only check is that data is not null, and since it wasn't initialized as null, this function returns SOAP_OK and
processing continues.

 soap_wsse_get_BinarySecurityToken(struct soap *soap, const char *id, char **valueType, unsigned char **data, int *size)
{
  _wsse__BinarySecurityToken *token = soap_wsse_BinarySecurityToken(soap, id);
  DBGFUN1("soap_wsse_get_BinarySecurityToken", "id=%s", id?id:"");
  if (token)
  {
    *valueType = token->ValueType;
    if (!token->EncodingType || !strcmp(token->EncodingType, wsse_Base64BinaryURI))
      *data = (unsigned char*)soap_base642s(soap, token->__item, NULL, 0, size);
    else if (!strcmp(token->EncodingType, wsse_HexBinaryURI))
      *data = (unsigned char*)soap_hex2s(soap, token->__item, NULL, 0, size);
    if (*data) <------ Passes as long as the uninitizlied pointer data points to a non-null
      return SOAP_OK;
  }
  return soap_wsse_fault(soap, wsse__SecurityTokenUnavailable, "BinarySecurityToken required");
}
```

Crash Information

```
Program received signal SIGSEGV, Segmentation fault.
(gdb) bt
#0  0x00007ffff771327b in ASN1_get_object () from /usr/lib/x86_64-linux-gnu/libcrypto.so.1.1
#1  0x00007ffff771a7a8 in ?? () from /usr/lib/x86_64-linux-gnu/libcrypto.so.1.1
#2  0x00007ffff771b98e in ?? () from /usr/lib/x86_64-linux-gnu/libcrypto.so.1.1
#3  0x00007ffff771c67d in ASN1_item_ex_d2i () from /usr/lib/x86_64-linux-gnu/libcrypto.so.1.1
#4  0x00007ffff771c6fb in ASN1_item_d2i () from /usr/lib/x86_64-linux-gnu/libcrypto.so.1.1
#5  0x00000000005d0b5e in soap_wsse_get_BinarySecurityTokenX509 (soap=0x7ffff7fbe010, id=<optimized out>) at ../../plugin/wsseapi.c:3161
#6  0x00000000005d5361 in soap_wsse_get_KeyInfo_SecurityTokenReferenceX509 (soap=0x7ffff7fbe010) at ../../plugin/wsseapi.c:4510
#7  0x00000000005d430b in soap_wsse_verify_Signature (soap=0x7ffff7fbe010) at ../../plugin/wsseapi.c:3812
#8  0x00000000005e2234 in soap_wsse_preparefinalrecv (soap=0x7ffff7fbe010) at ../../plugin/wsseapi.c:7659
#9  0x000000000057de14 in soap_end_recv (soap=0x7ffff7fbe010) at ../../stdsoap2.c:11512
#10 0x00000000005319a6 in soap_serve___wst__RequestSecurityToken (soap=0x7ffff7fbe010) at soapServer.c:95
#11 0x0000000000531695 in soap_serve_request (soap=0x7ffff7fbe010) at soapServer.c:62
#12 0x0000000000531343 in soap_serve (soap=0x7ffff7fbe010) at soapServer.c:37
#13 0x00000000004065bd in main (argc=2, argv=0x7fffffffe4b8) at wstdemo.c:204
0x00007ffff771327b in ASN1_get_object () from /usr/lib/x86_64-linux-gnu/libcrypto.so.1.1
```

Timeline

2020-11-05 - Vendor Disclosure

2020-12-16 - Vendor advised patch released on 2020-11-20

2021-01-05 - Public Release

CREDIT

Discovered by a member of Cisco Talos.

VULNERABILITY REPORTS

PREVIOUS REPORT NEXT REPORT

TALOS-2020-1188 TALOS-2020-1197