FusionAuth-SAMLv2 0.2.3 Message Forging

Authored by Felix Sieges

Posted Oct 2, 2020

Unauthenticated users can send forged messages to the FusionAuth to bypass authentication, impersonate other users or gain arbitrary roles. The SAML message can be send to the application without a signature even if this is required. The impact depends on individual applications that implement fusionauth-samlv2. Version 0.2.3 is vulnerable.

tags | exploit, arbitrary
advisories | CVE-2020-12676
SHA-256 | c0bc810aed6db58661b8cd13a1ebf5d20fed6fdb9c77567debaa3ab0cf809833 Download | Favorite | View

Related Files

Change Mirror Download

```
#################################################################
#
# COMPASS SECURITY ADVISORY
# https://www.compass-security.com/research/advisories/
#
#################################################################
#
# Product:  SAML v2.0 bindings in Java using JAXB
# Vendor:   FusionAuth
# CSNC ID:  CSNC-2020-002
# CVE ID:   CVE-2020-12676
# Subject:  Signature Exclusion Attack
# Risk:     High
# Effect:   Remotely exploitable
# Author:   Felix Sieges <felix.sieges@compass-security.com>
# Date:     2020-09-30
#
#################################################################

Introduction:
-------------
FusionAuth [1] provides authentication, authorization, and user management for any app.
SAML v2.0 bindings in Java using JAXB are a library used to integrate SAML
Authentication with Java Applications. Compass Security [2] identified a vulnerability
that allows remote attackers to forge messages and bypass authentication via a SAML
assertion that lacks a Signature element, aka a "Signature exclusion attack".


Affected:
---------
Vulnerable:
fusionauth-samlv2 0.2.3

Not vulnerable:
fusionauth-samlv2 0.2.4

Not tested:
No other version was tested, but it is believed for the older versions to be
vulnerable as well.


Technical Description
---------------------
Unauthenticated users can send forged messages to the FusionAuth to bypass
Authentication, impersonate other users or gain arbitrary roles. The SAML
Message can be send to the Application without a signature even if this is
required. The impact depends on individual applications that
implement fusionauth-samlv2.

The code which is responsible to verify the signature is called from the parseResponse
function [3]. The function checks whether a signature must be verified and if so
the function verifySignature is called to do signature verification checks.

https://github.com/FusionAuth/fusionauth-

```
  public AuthenticationResponse parseResponse(String encodedResponse, boolean verifySignature, PublicKey key)
      throws SAMLException {

    AuthenticationResponse response = new AuthenticationResponse();
    byte[] decodedResponse = Base64.getMimeDecoder().decode(encodedResponse);
    response.rawResponse = new String(decodedResponse, StandardCharsets.UTF_8);

    Document document = parseFromBytes(decodedResponse);
    if (verifySignature) {
      verifySignature(document, key);
    }
```


In the function verifySignature [4] the SAML message is parsed after validation that a
signature is attached to the message. If no signatures exist the function just returns.
Hence the parseReponse method assumes that the signature is valid and the SAML message
will be processed further.


```
  private void verifySignature(Document document, Key key) throws SAMLException {
    // Fix the IDs in the entire document per the suggestions at
http://stackoverflow.com/questions/17331187/xml-dig-sig-error-after-upgrade-to-java7u25
    fixIDs(document.getDocumentElement());

    NodeList nl = document.getElementsByTagNameNS(XMLSignature.XMLNS, "Signature");
    if (nl.getLength() == 0) {
      return;
    }

    for (int i = 0; i < nl.getLength(); i++) {
      DOMValidateContext validateContext = new DOMValidateContext(key, nl.item(i));
      XMLSignatureFactory factory = XMLSignatureFactory.getInstance("DOM");
      try {
        XMLSignature signature = factory.unmarshalXMLSignature(validateContext);
        boolean valid = signature.validate(validateContext);
        if (!valid) {
          throw new SAMLException("Invalid SAML v2.0 authentication response. The signature is invalid.");
        }
      } catch (MarshalException e) {
        throw new SAMLException("Unable to verify XML signature in the SAML v2.0 authentication response
because we couldn't unmarshall the XML Signature element", e);
      } catch (XMLSignatureException e) {
        throw new SAMLException("Unable to verify XML signature in the SAML v2.0 authentication response. The
signature was unmarshalled we couldn't validate it for an unknown reason", e);
      }
    }
  }
}
```

Workaround / Fix:

Upgrade to fusionauth-samlv2 0.2.4, where the signature checks are performed correctly

#TODO
Timeline:

2020-04-29:   Discovery by Felix Sieges
2020-05-06:   Assigned CVE-2020-12676
2020-05-06:   Initial vendor notification
2020-05-06:   Initial vendor response
```

File Archive: December 2022 <

Su	Mo	Tu	We	Th	Fr	Sa
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Top Authors In Last 30 Days

Red Hat 154 files
Ubuntu 73 files
LiquidWorm 23 files
Debian 18 files
malvuln 11 files
nu11secur1ty 11 files
Gentoo 9 files
Google Security Research 8 files
T. Weber 4 files
Julien Ahrens 4 files

File Tags

ActiveX (932)
Advisory (79,754)
Arbitrary (15,694)
BBS (2,859)
Bypass (1,619)
CGI (1,018)
Code Execution (6,926)
Conference (673)
Cracker (840)
CSRF (3,290)
DoS (22,602)
Encryption (2,349)
Exploit (50,359)
File Inclusion (4,165)
File Upload (946)
Firewall (821)
Info Disclosure (2,660)
Intrusion Detection (867)
Java (2,899)
JavaScript (821)
Kernel (6,291)
Local (14,201)
Magazine (586)
Overflow (12,419)
Perl (1,418)
PHP (5,093)
Proof of Concept (2,291)
Protocol (3,435)
Python (1,467)
Remote (30,044)
Root (3,504)
Ruby (594)
Scanner (1,631)
Security Tool (7,777)
Shell (3,103)
Shellcode (1,204)
Sniffer (886)

File Archives

December 2022
November 2022
October 2022
September 2022
August 2022
July 2022
June 2022
May 2022
April 2022
March 2022
February 2022
January 2022
Older

Systems

AIX (426)
Apple (1,926)
BSD (370)
CentOS (55)
Cisco (1,917)
Debian (6,634)
Fedora (1,690)
FreeBSD (1,242)
Gentoo (4,272)
HPUX (878)
iOS (330)
iPhone (108)
IRIX (220)
Juniper (67)
Linux (44,315)
Mac OS X (684)
Mandriva (3,105)
NetBSD (255)
OpenBSD (479)
RedHat (12,469)
Slackware (941)
Solaris (1,607)

```
2020-12-05:     Release of fixed Version
2020-09-30:     Coordinated public disclosure date


References:
-----------
[1] https://fusionauth.io/
[2] https://compass-security.com
[3] https://github.com/FusionAuth/fusionauth-
samlv2/blob/master/src/main/java/io/fusionauth/samlv2/service/DefaultSAMLv2Service.java#L502
[4] https://github.com/FusionAuth/fusionauth-
samlv2/blob/dbcf3ccabbc89f43ca322c1526a6a3589e3a1f1f/src/main/java/io/fusionauth/samlv2/service/DefaultSAMLv2Se
```


Spoof (2,166)
SQL Injection (16,102)
TCP (2,379)
Trojan (686)
UDP (876)
Virus (662)
Vulnerability (31,136)
Web (9,365)
Whitepaper (3,729)
x86 (946)
XSS (17,494)
Other

SUSE (1,444)
Ubuntu (8,199)
UNIX (9,159)
UnixWare (185)
Windows (6,511)
Other

Site Links
News by Month
News Tags
Files by Month
File Tags
File Directory

About Us
History & Purpose
Contact Information
Terms of Service
Privacy Statement
Copyright Information

Hosting By
Rokasec

Follow us on Twitter
Subscribe to an RSS Feed