OS HARDENING TECHNIQUES

Scenario

You are a cybersecurity analyst for yummyrecipesforme.com, a website that sells recipes and cookbooks. A former employee has decided to lure users to a fake website with malware.

The baker executed a brute force attack to gain access to the web host. They repeatedly entered several known default passwords for the administrative account until they correctly guessed the right one. After they obtained the login credentials, they were able to access the admin panel and change the website's source code. They embedded a javascript function in the source code that prompted visitors to download and run a file upon visiting the website. After embedding the malware, the baker changed the password to the administrative account. When customers download the file, they are redirected to a fake version of the website that contains the malware.

Several hours after the attack, multiple customers emailed yummyrecipesforme's helpdesk. They complained that the company's website had prompted them to download a file to access free recipes. The customers claimed that, after running the file, the address of the website changed and their personal computers began running more slowly.

In response to this incident, the website owner tries to log in to the admin panel but is unable to, so they reach out to the website hosting provider. You and other cybersecurity analysts are tasked with investigating this security event.

To address the incident, you create a sandbox environment to observe the suspicious website behavior. You run the network protocol analyzer tcpdump, then type in the URL for the website, yummyrecipesforme.com. As soon as the website loads, you are prompted to download an executable file to update your browser. You accept the download and allow the file to run. You then observe that your browser redirects you to a different URL, greatrecipesforme.com, which contains the malware.

The logs show the following process:

1. The browser initiates a DNS request: It requests the IP address of the yummyrecipesforme.com URL from the DNS server.
2. The DNS replies with the correct IP address.
3. The browser initiates an HTTP request: It requests the yummyrecipesforme.com webpage using the IP address sent by the DNS server.
4. The browser initiates the download of the malware.
5. The browser initiates a DNS request for greatrecipesforme.com.
6. The DNS server responds with the IP address for greatrecipesforme.com.
7. The browser initiates an HTTP request to the IP address for greatrecipesforme.com.

A senior analyst confirms that the website was compromised. The analyst checks the source code for the website. They notice that javascript code had been added to prompt website visitors to download an executable file. Analysis of the downloaded file found a script that redirects the visitors' browsers from yummyrecipesforme.com to greatrecipesforme.com.

The cybersecurity team reports that the web server was impacted by a brute force attack. The disgruntled baker was able to guess the password easily because the admin password was still set to the default password. Additionally, there were no controls in place to prevent a brute force attack.

Your job is to document the incident in detail, including identifying the network protocols used to establish the connection between the user and the website. You should also recommend a security action to take to prevent brute force attacks in the future.

Tcpdump traffic log

Number	Traffic log
1	14:18:32.192571 `IP your.machine.52444` > `dns.google.domain: 35084`+ A? `yummyrecipesforme.com.` (24)

2	14:18:32.204388 `IP dns.google.domain` > `your.machine.52444: 35084` 1/0/0 A `203.0.113.22` (40)
3	14:18:36.786501 `IP your.machine.36086` > `yummyrecipesforme.com.http`: `Flags [S]`, seq 2873951608, win 65495, options [mss 65495,sackOK,TS val 3302576859 ecr 0,nop,wscale 7], length 0
4	14:18:36.786517 `IP yummyrecipesforme.com.http` > `your.machine.36086`: `Flags [S.]`, seq 3984334959, ack 2873951609, win 65483, options [mss 65495,sackOK,TS val 3302576859 ecr 3302576859,nop,wscale 7], length 0
5	14:18:36.786529 `IP your.machine.36086` > `yummyrecipesforme.com.http`: `Flags [.]`, ack 1, win 512, options [nop,nop,TS val 3302576859 ecr 3302576859], length 0
6	14:18:36.786589 `IP your.machine.36086` > `yummyrecipesforme.com.http`: `Flags [P.]`, seq 1:74, ack 1, win 512, options [nop,nop,TS val 3302576859 ecr 3302576859], length 73: `HTTP: GET / HTTP/1.1`
7	14:18:36.786595 `IP yummyrecipesforme.com.http` > `your.machine.36086`: `Flags [.]`, ack 74, win 512, options [nop,nop,TS val 3302576859 ecr 3302576859], length 0
	...<a lot of traffic on the port 80>...

Number	Traffic log
8	14:20:32.192571 `IP your.machine.52444` > `dns.google.domain`: 21899+ A? `greatrecipesforme.com.` (24)
9	14:20:32.204388 `IP dns.google.domain` > `your.machine.52444`: 21899 1/0/0 A `192.0.2.17` (40)
10	14:25:29.576493 `IP your.machine.56378` > `greatrecipesforme.com.http`: `Flags [S]`, seq 1020702883, win

	65495, options [mss 65495,sackOK,TS val 3302989649 ecr 0,nop,wscale 7], length 0
11	14:25:29.576510 `IP greatrecipesforme.com.http` > `your.machine.56378`: `Flags [S.]`, seq 1993648018, ack 1020702884, win 65483, options [mss 65495,sackOK,TS val 3302989649 ecr 3302989649,nop,wscale 7], length 0
12	14:25:29.576524 `IP your.machine.56378` > `greatrecipesforme.com.http`: `Flags [.]`, ack 1, win 512, options [nop,nop,TS val 3302989649 ecr 3302989649], length 0
13	14:25:29.576590 `IP your.machine.56378` > `greatrecipesforme.com.http`: `Flags [P.]`, seq 1:74, ack 1, win 512, options [nop,nop,TS val 3302989649 ecr 3302989649], length 73: `HTTP: GET / HTTP/1.1`
14	14:25:29.576597 `IP greatrecipesforme.com.http` > `your.machine.56378`: `Flags [.]`, ack 74, win 512, options [nop,nop,TS val 3302989649 ecr 3302989649], length 0
	...<a lot of traffic on the port 80>...

How to read the tcpdump traffic log

This reading explains how to identify the brute force attack using tcpdump.

Action	Log entry
DNS resolution request	14:18:32.192571 (a) `IP your.machine.52444` (b) > `dns.google.domain: 35084`+ (c) A? `yummyrecipesforme.com.` (d) (24) a: Timestamp b: Source computer with port (52444) c: DNS server

	d: Destination URL
Reply to DNS resolution request	14:18:32.204388 `IP dns.google.domain` > `your.machine.52444: 35084` 1/0/0 A `203.0.113.22` (a) (40) a: IP address of the destination URL
Connection request [S]	14:18:36.786501 `IP your.machine.36086` > `yummyrecipesforme.com.http`: `Flags [S]`, seq 2873951608, win 65495, options [mss 65495,sackOK,TS val 3302576859 ecr 0,nop,wscale 7], length 0
Reply to connection request [S.]	14:18:36.786517 `IP yummyrecipesforme.com.http` > `your.machine.36086`: `Flags [S.]`, (a) seq 3984334959, ack 2873951609, win 65483, options [mss 65495,sackOK,TS val 3302576859 ecr 3302576859,nop,wscale 7], length 0 a: [S.]: Connection request acknowledged
TCP flag codes	`Flags [S]` - Connection Start `Flags [F]` - Connection Finish `Flags [P]` - Data Push `Flags [R]` - Connection Reset `Flags [.]` - Acknowledgment
Browser downloading malware	14:18:36.786589 `IP your.machine.36086` > `yummyrecipesforme.com.http`: `Flags [P.]`, (a) seq 1:74, ack 1, win 512, options [nop,nop,TS val 3302576859 ecr 3302576859], length 73: `HTTP: GET` (b) / `HTTP/1.1` (c) a: Data push acknowlegded b: The browser is requesting data from yummyrecipesforme.com with the HTTP: GET method c: using HTTP protocol version 1.1

Security incident report

Section 1: Identify the network protocol involved in the incident
The network protocol involved in the incident is the HTTP protocol, commonly associated with port 80, as it is used for unencrypted HTTP traffic. Since the issue occurs when accessing the legitimate website's web server, it is known that these requests to web servers for internet pages are made via the HTTP protocol. By analyzing the traffic logs from tcpdump, it can be observed that a file is downloaded using an HTTP method, which redirects the user to the website containing the malware.

Section 2: Document the incident
An attacker gained access to the web host of the company yummyrecipesforme.com through a brute force attack. With administrator credentials, they modified the website's source code, inserting a JavaScript function that prompted visitors to download and run a file upon entering the site. That file redirected the clients to a fake version of the website that contains the malware. Several hours after the attack, some customers complained that the site had asked them to download a file to update the browser. Customers reported that after executing the file, the web address changed, and their computers began to slow down. The cybersecurity analysts team, in a test environment, observed the suspicious behavior of the site: upon accessing yummyrecipesforme.com, they were prompted to download a file to "update the browser." Running the file redirected the browser to a fake version of the site, greatrecipesforme.com, which contained the malware. Activity logs confirmed the attack. The browser made DNS and HTTP requests that first led to yummyrecipesforme.com and then, once the file was downloaded, redirected to greatrecipesforme.com, revealing the malicious nature of the code. A senior analyst reviewed the source code and verified that the JavaScript

injected into the compromised site triggered the download of the infected file and the redirection. Analysis of the downloaded file showed a script that automated redirection to the fake version of the site, confirming the malicious manipulation of the original website.

Section 3: Recommend one remediation for brute force attacks
To protect against brute force attacks, several recommendations can be made: 1. Do not allow the use of previously used passwords. Since the attacker exploited this vulnerability by testing several default passwords, it is important to prevent old passwords from being used as the default. 2. Establish frequent password updates. 3. Enable two-factor authentication (2FA). This provides two forms of authentication, both a password and a one-time passcode sent to an email or phone. 4. Finally, login attempts can also be limited, as a brute force attack involves testing multiple password combinations.