ZAProxy

ZAProxy (https://www.owasp.org/index.php/ZAP; https://github.com/zaproxy/zaproxy) és un proxy HTTP/S (és a dir, un programa intermediari entre un client i un servidor web) desenvolupat pel col.lectiu OWASP; és similar a la coneguda -però privativa- eina anomenada Burp Suite (https://portswigger.net/burp). El seu objectiu principal és inspeccionar/manipular el tràfic HTTP/S que ha sigut redireccionat cap a ell, però pot fer moltes altres coses, com ara descobrir nous servidors web i subdirectoris dins d'ells (el que a vegades s'anomena "spider" o "crawling"), realitzar atacs de força bruta o amb diccionari en formularis de login (o, en general, realitzar atacs de "fuzzing" a qualsevol tipus d'entrada), detectar (i sovint explotar) vulnerabilitats webs, etc.

NOTA: Una altra alternativa (també lliure és Pownjs (https://pownjs.com), la qual, a més de ser HTTP/S proxy, incorpora igualment un descobridor de hosts i subdirectoris, un atacant de contrasenyes web i un escanejador de vulnerabilitats

NOTA: Existeixen altres eines específiques que només s'encarreguen d'una funcionalitat concreta que proporciona ZAProxy. Per exemple, de descobridors de subdirectoris en servidors web en tenim les eines "Dirb" (obsoleta) o també:

- DirBuster (https://github.com/KajanM/DirBuster)
- DirSearch (https://github.com/maurosoria/dirsearch)
- Pathbrute (https://github.com/milo2012/pathbrute)
- GoBuster (https://github.com/OJ/gobuster)

De provadors de contrasenyes online en tenim, per exemple:

- THC-Hydra (https://github.com/vanhauser-thc/thc-hydra)
- Medusa (http://h.foofus.net/?page_id=51)
- Ncrack (https://nmap.org/ncrack)

D'escanejadors de vulnerabilitats en tenim, per exemple:

- Nikto (https://cirt.net/Nikto2)
- Nmap (https://nmap.org/nsedoc)
- OpenVAS (https://github.com/greenbone/openvas-scanner)
- WPScan (https://wpscan.org) -Especialitzat en WordPress-

Etc, etc

NOTA: A més existeixen altres eines especialitzades en explotar vulnerabilitats específiques, com ara SQLmap (http://sqlmap.org) o Commix (https://commixproject.com) , de les quals en parlarem més endavant.

Configuració prèvia

Per fer que el tràfic generat per qualsevol navegador pugui ser interceptat per ZAProxy, primer hem de canviar la configuració d'aquest navegador per tal de què redireccioni tot aquest tràfic al proxy desitjat. Al Firefox, concretament, això es fa anant a "Preferències->General" i, a la secció "Paràmetres de xarxa", pulsant sobre el botó "Paràmetres": al quadre que apareix es pot indicar manualment la IP i port on estarà escoltant el proxy en qüestió peticions de tipus HTTP (i també HTTPS).

NOTA: Al quadre de configuració del proxy usat pel navegador també apareix l'opció d'usar el "proxy del sistema". Això vol dir que no s'estarà indicant explícitament cap proxy allà sinó que es farà servir el que s'hagi indicat al panel de configuració de Gnome, apartat "Xarxa, opció "Servidor intermediari de xarxa"

Per defecte, ZAProxy escolta a través de la IP 127.0.0.1 i al port 8080. Si es vol canviar això (per exemple, si es vol interceptar tràfic generat per navegadors de màquines remotes, cosa que amb la IP "loopback" no es pot fer), es pot anar al menú "**Tools->Options**" de ZAProxy i buscar el submenu "**Local proxy**".

Cal saber, però, que si volem que ZAProxy pugui observar (i manipular) tràfic HTTPS, serà necessari realitzar un pas més. Aquest pas és el mateix que vam realitzar per aconseguir que Bettercap també pogués intervenir tràfic HTTPS: és a dir, crear un certificat arrel d'autoritat certificadora corresponent al propi ZAProxy i "incrustar-lo" a la configuració del navegador per a què aquest confïi sense reserves en tots els certificats dels servidors web visitats que hagi rebut de part de ZAProxy, certificats que estaran signats "al vol" per aquest certificat arrel "impostor". La primera vegada que es posa en marxa ZAProxy es genera automàticament aquest certificat arrel, així que l'únic que haurem de fer és exportar-lo en forma de fitxer i copiar aquest fitxer a tots els sistemes on volguem importar-lo al navegador corresponent.

Això es fa de la següent manera: cal anar al menú "**Tools->Options**" de ZAProxy i buscar el submenú "**Dynamic SSL Certificates**": al quadre que apareix caldrà clicar sobre el botó "Save" (també hi apareix un botó "Generate" per si es vol regenerar un nou certificat arrel) i seguidament "OK". El fitxer resultant, anomenat "owasp_zap_root_ca.cer", és el fitxer que caldrà importar al navegador de la maneixa manera que quan vam veure Bettercap (és a dir, anant al menú "Preferències->Privadesa i seguretat->Certificats->Mostra els certificats->Entitats->Importa" i dient que es vol confiar en aquest certificat arrel per navegar per la web). Instruccions més detallades sobre aquest procés les podeu trobar a https://github.com/zaproxy/zap-core-help/wiki/HelpUiDialogsOptionsDynsslcert#generate

Amb aquestes dues opcions (IP i port d'escolta i generació/distribució de certificat arrel) ja podrem inspeccionar el tràfic HTTP/S de qualsevol navegador de la nostra LAN. No obstant, a vegades només voldrem realitzar estudis ràpids sobre aspectes concrets d'aquest tràfic i ens "valdrà la pena" manipular la configuració dels navegadors reals per això: en aquests casos, ZAProxy incorpora un navegador propi ja preconfigurat (és a dir, amb el tràfic ja "proxificat" i amb el certificat arrel importat) per fer-lo servir en comptes dels navegadors reals. Es pot executar anant al menú "**Tools->Launch the ZAP JxBrowser**" de ZAProxy.

> NOTA: Si en comptes d'usar el JxBrowser es volgués executar un altre navegador igualment preconfigurat però dels instal.lats als sistema, es pot escollir el navegador en qüestió al desplegable que apareix al costat del botó "Launch Browser" dins de la pestanya "Quick Start" que es mostra a la part central dreta de la finestra de ZAProxy (o bé, alternativament, mitjançant un botó de la seva barra de botons).

Preparació (opcional)

When you've got your proxy set up it's often useful to create a new "context". Contexts are a way to group relevant URLs, so that ZAP only shows you the traffic you care about. In other words: contexts are a way of relating a set of URLs together: you can define any contexts you like, but it is expected that a context will correspond to a web application. Create a new context by clicking on the "new context button" below "Sites" tab, and giving it a name. Then add addresses to this new context by navigating to "File -> Session properties" and opening the "Contexts" sub-menu. From there, select the context you created and find the "Include in context" tab; finally, add the URLs you're filtering for in the menu there. Alternatively, if the hosts already are listed in your "Sites" tab in ZAP, you can right click and select add to context.

To filter on a configured context, you want to mark it "in scope" and likely mark the "Default Context" as "not in scope". Do so by right clicking the contexts and selecting add to scope or remove from scope as required. Additionally, you must also enable scope filtering in the various lists you see in the ZAP UI by clicking the little bulls-eye symbol

> NOTA: ZAProxy has four modes, which can be changed via the toolbar and is persisted between sessions:
> **Safe mode* : This mode doesn't allow you to do anything that is potentially dangerous.
> **Protected mode* : This mode allows you simulate potentially dangerous vulnerabilities. User can only perform harmful actions on URLs which are mentioned in the scope.
> **Standard mode* : In this mode user can do anything that is relevant.
> **ATTACK mode* : New nodes in scope are actively scanned as soon as they are discovered. We can define the scan policy to be used for this mode in the Options Active Scan screen.
> It is strongly recommended that you use the 'Protected mode' to ensure that you only attack sites that you mean to. Things that will not be possible in either 'Safe mode' or 'Protected mode' when not acting on URLs in the Scope are: "Spider – crawling", "Active Scanning", "Fuzzing", "Force Browsing", "Breaking (intercepting)" and "Resending requests"

Funcionalitats

Once browser start to navigate through Internet and pages load, go back to the ZAProxy window. All of the requests and responses made in order to load the pages will have been intercepted and forwarded by ZAProxy so they should show up in a table below the **History** tab, in the bottom of the window. You can see in this table the "HTTP Method", "URL", "RTT", "Size response body" columns (among others) showing these main details about every request and response. Also you can filter the lines shown in the table with the button that appears above the table with a funnel picture: the filter can be done by method, by HTTP code response and/or URL regular expressions, among other options. You can export table content to a CSV file,

too, using the corresponding button that appears next to the filter button. But the most interesting feature here is that if you select any of the lines in table below the "History" tab, you will see all the related request and response data it handles (mainly the request and response headers and body) in respective "**Request**" and "**Response**" tabs (located next to "Quick start" tab, on main right panel of ZAProxy).

> NOTA: Instead of having a "Request" tab and a different "Response" tab on main ZAProxy window, you can combine both to see in the same time all the related information (request's header and body and response header and body). This can be done clicking on this button: 

Now that we can record traffic between a client and a server , you can use the breakpoints feature of ZAProxy to stop a request in-flight and modify it. To do so, firstly you need to right click on a request in the history tab (or, more generally, in the URL below "Sites" list) and locate the **Break…** option. In general, using the default URL's regular expression shown in the emerging window is fine but you can change it (you could use only a matching string instead of a regular expression if you select "Contains" instead of "Regex", or inverse the matching hit, etc) or even select another matching criteria, like request header, request body, response header or response body. Once you set up and save the break point you will see a new "**Break points**" tab in the bottom panel and a new tab in the main panel called "**Break**". Then you should use your web client to make the request to the desired destination server again: the main "Break" tab will let you modify then the request headers and body before sending it to that server; only when you press "**Play**" button (located on buttons bar) this (modified or not) request will arrive to the server. Inmediately after this, ZAProxy will show the response headers and body, which you'll able to modify to again; only when you press "Play" button again this (modified or not) response will arrive to browser.

> NOTA: The "Play" button represents a "continue" button: it breaks when it arrives to another break point. There's another button on its left (the one with the bar and the triangle sign) which represents a "step" button: it breaks at every intercepted request and response

Another interesting feature of ZAProxy is its "spider-crawler" (similar to Burp's one). Un "crawler" o "spider" es una herramienta, o en este caso una funcionalidad de ZAProxy, que sirve para identificar los enlaces existentes en un target; de esta manera, nos podemos hacer una idea de la manera en la que está compuesta el sitio a analizar e identificar posibles directorios o archivos sensibles que nos pueden ser útiles a la hora de nuestra auditoría. La forma en la que ZAProxy trabaja es recursiva; es decir, que a medida que encuentra nuevos enlaces, los va siguiendo, identificando así *href*, *src*, *http-equiv* o *location* entre otros atributos de html, *get* y *post* en lenguajes dinámicos e incluso los vínculos que están escondidos de los bots de indexación en el fichero "robots.txt". Esto nos da la posibilidad de hacer el crawling bastante granular, ya que todo es seteable desde las opciones de configuración. To run it you need to right click on a request in the history tab (or, more generally, in the URL below "Sites" list) and select the "**Attack->Spider**" option. An emerging window will appear showing the starting URL to do the crawling (you can change it but in general defaults are fine...pay attention to "Spider subtree only" option). Once you set up it you will see a new "**Spider**" tab in the bottom panel and scan will begin. In that tab you will see the process of running the scan and its results; you'll be able to stop/pause when you want, too. This tool can be configured in the "**Tools → Options → Spider**" window: there we can establish the number of concurrent processes, enable/disable the crawler in files' metadata or in site's comments, among other options.

Another interesting feature of ZAProxy is its integrated active vulnerability scanner (similar to Burp Pro´s one), being a great complement of Nikto, Nmap or OpenVAS virtues, thus. To run it you need to right click on a request in the history tab (or, more generally, in the URL below "Sites" list) and select the "**Attack->Active scan**" option. An emerging window will appear showing the starting URL to do the crawling (you can change it but in general defaults are fine). Once you set up it you will see results on "**Alerts**" tab (located in the bottom panel). This scanner can be configured in the "**Tools → Options → Active scan**" window: there we can configure the number of concurrent hosts, concurrent threads, timeouts...

On the other hand, ZAP can do also a passive scan: this kind of scan only intercepts server's responses and isn't intrusive. Its rules can be configured in "**Tools → Options → Passive scan**" window: there we can choose the rules to use, edit them, create new ones or delete them. Las reglas por defecto incluidas en el escaner pasivo de ZAP incluyen la capacidad de detectar comentarios, direcciones de correos electrónicos, cookies, formularios, objetos, contraseñas, scripts, campos ocultos, entre otras.

El "Forced Browse" es un tipo de ataque para forzar la navegación dentro de un dominio mediante fuerza bruta de nombres de subdirectorios y ficheros con el fin de identificar recursos que no son accesibles desde una referencia (eso lo haría un "crawler") pero aun están en algun directorio dentro del web server. To do this you must select a specific URL from History (or, more generally, in the URL below "Sites" list) and select one of the following options shown in (right-clicked) contextual menu: "**Attack->Forced Browse Site**" (es utilizado para la identificación de contenido no vinculados en el directorio del dominio, se ejecuta por defecto o seteando nuestro propio diccionario); "**Attack->Forced Browse Directory**" (cumple la misma función que el anterior, la diferencia es que identifica sobre un directorio y no sobre todo el dominio) o "**Attack->Forced Browse Directory (and children)**" (igual al anterior, sólo que además también identifica el contenido de los subdirectorios). Para definir las opciones de configuración y establecer nuestro propio diccionario de fuerza bruta en la herramienta debemos ir al menú superior "**Tools -> Options -> Forced Browse**" Cuando lancemos el ataque, podemos elegir si queremos utilizar el diccionario definido por defecto o buscar y seleccionar el que deseamos emplear. Una vez lanzado, en el panel inferior veremos como comienzan a desplegarse las peticiones que hace ZAProxy sobre el directorio seleccionado y la respuesta de cada request. De esta manera podemos identificar, por ejemplo, información sensible, backdoors shell en php con nombres comunes como c99.php o DAws.php, etc que generalmente no tienen ninguna referencia o hipervínculo desde el dominio pero si están presentes dentro del webserver.

Another interesting ability of ZAProxy is doing "fuzz" attacks, which are similar to the Burp's "Intruder" ones. El fuzzing es una técnica mediante la cual se puede comprobar la forma en la que responde, en éste caso una web application, ante el ingreso de datos aleatorios o secuenciales para para identificar directorios o archivos, detectar vulnerabilidades de inyección de código e incluso para realizar validaciones de contraseña por fuerza bruta. To do this you must select a specific URL from History (or, more generally, in the URL below "Sites" list) and select "**Attack→Fuzz**" option shown in (right-clicked) contextual menu. An emerging window will appear asking for the specific characteristic of the attack: we will see some of them in exercices. Once you set up it you will see results on "**Fuzzer**" tab (located in the bottom panel). This scanner can be configured in the "**Tools → Options → Fuzzer**" window.

> NOTA: Si no queremos realizar un ataque de forma sensata y "elegante" (es decir, identificando el target con un escaneo pasivo y focalizando en lo que nos interesa con un escaneo activo) sino que optamos por tirarle a todo de manera cavernícola, tenemos una pestaña llamada "Quick start", donde escribiendo la URL a atacar y clicando en el botón correspondiente no sólo ejecutaremos un scanning activo, sino también el "spider", el "fuzzer" y el "brute forcer" (y saltarán todas las alarmas, claro)

EXERCICIS:

1.-a) Vés a https://github.com/zaproxy/zaproxy/wiki/Downloads i descarrega (a la teva màquina real) la versió "ZAP Weekly". Després de descomprimir el paquet resultant, fes clic sobre l'arxiu "zap.sh": s'haurà de posar en marxa el programa. Vés al menú "Tools->Launch ZAP JxBrowser" per obrir un navegador que generi un tràfic HTTP/S interceptable per ZAProxy.

NOTA: Per a què ZAP funcioni a la màquina en qüestió s'ha de tenir instal.lat prèviament el "runtime environment" de Java

b) Escriu a la barra de direccions d'aquest navegador la URL http://httpstat.us . ¿Què veus a la pestanya "History"? ¿Què veus a les pestanyes "Request" i "Response" si sel.lecciones alguna línia de les que apareixen sota la pestanya "History"?

c) Crea un break point sobre la URL http://httpstat.us i a continuació escriu a la barra de direccions d'aquest navegador la URL http://httpstat.us/200 ¿Què veus a la pestanya "Request"? ¿Apareix alguna línia nova sota la pestanya "History"? Clica sobre el botó de "Play". ¿Què veus a la pestanya "Response"? ¿Apareix alguna línia nova sota la pestanya "History"?

d) Mantenint el mateix "break point" de l'apartat anterior, escriu ara a la barra de direccions del navegador la URL http://httpstat.us/404 però ara, abans de clicar sobre el botó "Play", modifica el valor de la petició mostrada a la pestanya "Request" per a què aparegui així: *GET http://httpstat.us/**501** HTTP/1.1* (també pots modificar el valor d'alguna capçalera de client, si vols). Un cop modificat el valor indicat, clica finalment sobre el botó "Play". ¿Quina resposta obtens? Per què? ¿Què veus al navegador?

e) Elimina el "break point" fet servir als apartats anteriors. Ara escriu a la barra de direccions del navegador la URL http://httpstat.us/403 , sel.lecciona la línia corresponent a aquesta petició<->resposta sota la pestanya "History" i seguidament, sobre aquesta mateixa línia (o bé sobre el contingut de la pestanya "Request") clica amb el botó dret i sel.lecciona l'opció del menú contextual "Open/Resend with Request Editor...". ¿Què mostra el quadre emergent que apareix? Modifica en aquest nou quadre el valor de la petició per a què ara sigui *GET http://httpstat.us/**501** HTTP/1.1* (també pots modificar el valor d'alguna capçalera de client, si vols) i envia-la. ¿Quina resposta obtens?

f) Imagina que vols atacar un servidor web que ofereix un formulari web que permet pujar imatges (és a dir, fitxers de tipus .jpg, .png i .gif). Imagina que en voler pujar a través d'aquest formulari un "payload2 de tipus "reverse-php" generat per msfvenom obtens un missatge d'error advertint que només s'admeten pujades d'aquests tres tipus d'imatges. Sabent que la detecció del tipus de fitxer a pujar podria realitzar-se pel servidor mitjançant la inspecció de la capçalera de client "Content-Type", detalla quins passos hauries de fer en ZAProxy per interceptar la pujada, modificar el valor de la capçalera en qüestió per un de vàlid i reprendre la pujada per comprovar si el servidor admet llavors el fitxer maliciós.

2.-a) Inicia una nova sessió de ZAProxy. Obre el navegador propi de ZAProxy i inicia logueja't en https://www.instagram.com (o en https://www.facebook.com o en alguna altra xarxa social on tinguis compte d'usuari personal). Utilitza la pestanya "Search" per localitzar la petició corresponent a l'inici de sessió (pots buscar per exemple pel teu nom d'usuari) i, un cop trobada, observa a la pestanya "Request" en quin lloc de la petició apareixen el nom d'usuari i contrasenya introduïts. ¿Com combinaries el que acabes de descobrir amb Bettercap?

b) Inicia una nova sessió de ZAProxy. Crea un nou "break point" general (clicant sobre el botó amb la icona de "X" vermella) que detecti un "Response Header" amb el valor de "application/javascript" (se suposa que per la capçalera "Content-Type"). Obre el navegador propi de ZAProxy i vés a https://www.elpais.com ; a cada resposta de tipus Javascript interceptada sel.lecciona tot el seu cos (CTRL+A a la pestanya "Response"), esborra'l (SUPR) i passa a la següent (botó "Play"); així fins que hagis eliminat totes les respostes de tipus Javascript. Compara la pàgina web final vista d'aquesta manera respecte la pàgina web mostrada en el navegador Firefox de la màquina real (on el Javascript no ha estat eliminat). ¿Quines diferències veus? ¿I si en comptes d'esborrar tot el contingut Javascript de les respostes, el substituïssis per la línia *alert("Hola");* ¿què passa?

c) Sota la pestanya "History" apareix la columna "Tags", que informa de contingut interessant a les respostes obtingudes de cada petició. Aquest contingut és reconegut gràcies a les regles indicades a "Tools->Options->Passive Scan Tags". ¿Com podries fer per ZAProxy reconegués l'existència d'enllaços "<img" (és a dir, imatges)? Prova-ho amb una web qualsevol. ¿I per saber si una determinada resposta conté algun enllaç a "google-analytics"?

3.-a) Inicia una nova sessió de ZAProxy. Obre el navegador propi de ZAProxy i vés a http://scanme.nmap.org Descriu amb les teves paraules dues alertes (a escollir) de les que ZAProxy hagi trobat per aquest servidor web mostrades a la pestanya "Alert".

b) Llença sobre aquesta web (sel.lecciona-la de "Sites") un atac "Spider". ¿Què veus sota la pestanya "Spider" que apareix al panell inferior de ZAProxy (i sota la pestanya "Sites")? ¿A aparegut alguna alerta més sota la pestanya "Alerts"?

c) Seguidament, llença sobre aquesta web un atac "Active scan". ¿Què veus sota la pestanya "Active scan" que apareix al panell inferior de ZAProxy? Obre el quadre "Show scan progress details" per tenir més informació del procés d'escaneig (aquest quadre s'obre amb un botó situat a la dreta del botó de parar). ¿A aparegut alguna alerta més sota la pestanya "Alerts"?

4.-a) En aquest exercici intentarem identificar una vulnerabilitat d'injecció SQL mitjançant "fuzzing". Inicia una nova sessió de ZAProxy. Obre el navegador propi de ZAProxy i vés a http://webscantest.com

aII) Realitza-hi un atac "Spider"; seguidament localitza l'arxiu "search_by_name.php" sota la carpeta "infodb" i clica-hi a sobre amb el botó dret per sel.leccionar l'opció "Atack->Fuzz...".

aIII) Al quadre que apareixerà podries sel.leccionar la capçalera de la petició (de tipus POST) o bé el seu cos segons on vulguis introduïr la "càrrega" (payload) a provar contra el lloc (és a dir, la llista de valors que es llençaran un darrera l'altre contra el destí per veure com reacciona). En aquest exercici has de posicionar el cursor al final del cos de la petició (perquè afegirem les càrregues a partir de la seva posició) i tot seguit clica sobre el botó "Add" per configurar els "payloads" per provar contra el lloc.

aIV) Al quadre que apareixerà hauràs de sel.leccionar novament "Add" per veure la llista de "payloads" disponibles a llençar contra el lloc web. ¿Quin tipus de "payload" és el de tipus "Strings"? ¿I el de tipus "Numberzz"? ¿I de tipus "File"?

aV) Sel.lecciona els "payloads" de tipus "File fuzzers" i sel.lecciona "jbrofuzz" (que és una eina interna d'OWASP que ofereix llistes pregenerades de valors de molts tipus). Dins de les seves opcions escull "Injection". ¿Què mostra l'apartat "Payload preview"? Accepta.

aVI) Clica sobre el botó "Start Fuzzing": ZAProxy començarà llavors a comprovar un a un els payloads imprimint a pantalla la resposta a cadascun d'ells sobre el destí. Si veus que algun payload en concret retorna "OK" en comptes de "Reflected", podràs anar al lloc web i escriure la cadena en qüestió (columna "Payloads") per obtenir el contingut de la base de dades.

aVII) Observa (a la pestanya "Request") el valor del cos de les diferents peticions realitzades durant l'apartat anterior. Si ara tornes a realitzar l'apartat aV) però escollint com a payload "jbrofuzz->Number systems" i seguidament tornes a realitzar l'atac, ¿quin valor veus llavors al cos de les peticions realitzades?

b) Sense tancar la sessió ZAProxy actual, vés a la pestanya "Search" del panell inferior de ZAProxy i busca "login.php". Sel.lecciona la petició POST corresponent i fixa't en el contingut del cos que apareix sota la pestanya "Request". Clica-hi a sobre amb el botó dret del ratolí i escull l'opció "Fuzz...".

bII) Sel.lecciona amb el ratolí el valor del paràmetre "login=" que apareix al quadre emergent i tot seguit clica al botó "Add"->"Add". Crea un payload de tipus "Strings" amb el contingut següent (un valor per línia): "admin", "root", "user" i "testuser".

bIII) Sel.lecciona amb el ratolí el valor del paràmetre "passwd=" que apareix al quadre emergent base i tot seguit clica al botó "Add"->"Add". Crea un payload de tipus "Numberzz" amb un contingut que vagi del número 1 al 1000 d'un en un.

bIV) Clica sobre el botó "Start Fuzzing": ZAProxy començarà llavors a comprovar els payloads (cada nom d'usuari indicat amb tot el conjunt de valors de contrasenyes possibles) imprimint a pantalla la resposta de la prova de cadascun d'ells sobre el destí. Pots observar també (a la pestanya "Request") el valor del cos de les diferents peticions realitzades durant l'apartat anterior. ¿Aconsegueixes trobar el login bo?

5.- a) ¿Què obtens si sel.lecciones l'opció del menú "Report ->Generate HTML Report" de ZAProxy?

> NOTA: En realidad, más que los informes ("reports"), lo que compararemos son las sesiones. De esta manera, si identificamos vulnerabilidades que deben ser remediadas, cuando auditamos nuevamente podemos corroborar con la sesión (y el informe también) si lo los responsables han solucionado las vulnerabilidades ya informadas o si siguen presentes a la espera de un atacante externo. Por eso en el menú 'Report' tenemos la opción de "Comparte with another session", la cual nos abre una ventana para poder elegir la sesión a comparar previamente guardada en el disco duro.

b) ¿Per a què serveix l'opció de menú "Tools-> Encode/Decode/Hash" ?

c) ¿Per a què serveix l'opció de menú "Tools-> Import a file containing URLs" ?

d) Escull l'opció del menú "View->Show all tabs" i vés a la pestanya "Port Scan". Selecciona un "host" qualsevol i inicia l'escaneig. ¿A quina altra eina molt coneguda de terminal seria similar aquesta pestanya?

6.-a) Clica sobre el botó que apareix a la barra de botons de ZAProxy mostrant una icona amb tres quadradets: un vermell a l'esquerra, un verd a la dreta i un blau a sobre dels anteriors (o alternativament, vés al menú "Help->Check for updates"). Veuràs que apareix un quadre amb una pestanya mostrant tots els "plugins" de ZAProxy que hi ha instal.lats (i que es poden actualitzar, si s'escau), els quals ofereixen cadascun una determinada funcionalitat específica, i una altra pestanya mostrant tots els "plugins" que es poden descarregar del repositori central de ZAProxy.

> NOTA: També es poden descarregar manualment aquests plugins des de https://github.com/zaproxy/zap-extensions/releases i instal.lar-los manualment anant al menú "File->Load Add-on File..."

b) Instal.la el plugin "Call graph". Inicia una nova sessió de ZAProxy. Obre el navegador propi de ZAProxy i vés a la pàgina web del centre, i entretent-te navegant per alguns dels seus enllaços ("Departaments", "Cicles formatius", etc). Finalment, sota la pestanya "Sites" clica amb el botó dret sobre la URL de la web en qüestió i sel.lecciona l'opció "Call graph->One site". ¿Què veus?

c) ¿Per a què serveix el plugin "Fuzzdb files"? Digues per a què serveixen altres dos plugins que et cridin l'atenció

d) Instal.la manualment el plugin https://github.com/h3xstream/burp-retire-js (no disponible des del quadre d'actualitzacions de ZAProxy) i digues què fa