**Identificación de características**

Las identificación de características es un paso muy importante en la detección de ataques, pues de la correcta identificación de estas depende el éxito que tendrá el clasificador utilizado. Estas características describen el comportamiento de un ciberataque, y plasman la interacción que el ciberataque tiene en sus distintas fases tanto en la capa de aplicación como en la capa de red.

Se analizarán distintos propuestas en las cuales se evalúan diferentes características para la identificación de ciberataques.

**Para peticiones HTTP**

El autor en [1] propone una serie de características para el protocolo HTTP que pueden ser útiles al momento de detectar ataques tales como *XSS, SQL Injection, DoS,* entre otros.

* **Información general de la petición:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Descripción** | **Tipo** | **Ataque asociado** |
| Req-length | Longitud de la petición | Entero positivo | Ataques de Buffer Overflow |
| URI-length | Longitud URI | Entero positivo | Buffer Overflow, URI decoding |
| Req-method | Tipo de la petición (GET, POST, HEAD, …) | Simbólico | - |
| Req-resource-type | Tipo de recurso solicitado (html, asp, php, …) | Nominal | - |
| Num-param | Número de parámetros | Entero positivo | Validación de entradas |
| Num-arg | Número de argumentos | Entero positivo | Validación de entradas |
| Is-req-correct | ¿La petición se ajusta al protocolo HTTP? | Booleano | Anomalías de la URL, URL decoding |

* **Contenido de la petición:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Descripción** | **Tipo** | **Ataque asociado** |
| Num-NonPrintChars | Número de caracteres especiales y shell codes en la petición HTTP (x86, carriage return, punto y coma,...) | Entero positivo | Buffer Overflow, codigos shell, URL decoding |
| SQL-cmd-tricks | ¿La petición contiene comandos SQL (“--,OR 1==1, …)? | Booleano | Inyección SQL |
| Shell-cmds | ¿La petición contiene comandos Shell (Todos los comandos shell de los sistemas operativos)? | Booleano | Inyección de comandos |
| Sensitive-files | ¿La petición contiene referencias a información sensible (etc/passwd, …)? | Booleano | Fuga de información, accesos no autorizados |
| Directory-traversal | ¿La petición contiene trucos de recorrido de directorio (presencia de token como “../”, …)? | Booleano | Recorrido de directorio |
| Oversized-values | ¿La petición contiene potencialmente valores numéricos sobredimensionados? | Booleano | Malinterpretación del valor |
| Default-login-passwd | ¿La petición incluye usuarios y contraseñas predeterminados de fábrica (guest, anonymous, root, admin, …? | Booleano | Ataques de diccionario, fuerza bruta, ... |
| Script-injection | ¿La URI contiene un tag de script (“<script”, “<meta”, …)? | Booleano | Cross Site Scripting |

* **Respuesta del servidor:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Descripción** | **Tipo** | **Ataque asociado** |
| Resp-code | Código de respuesta de la petición HTTP (200, 404, 500, …) | Nominal | - |
| Is-html-Response | ¿La respuesta es un archivo html? | Booleano | - |
| Response-time | Tiempo transcurrido desde la correspondiente petición HTTP | Real | DoS |
| Script-type | El tipo de script incluido en la respuesta (Java, Visual Basic, …) | Nominal | Cross Site Scripting |
| Writing-script | ¿El flujo de respuesta incluye script de funciones de escritura (document.write(), …)? | Booleano | Fijación de ID de sesión, ... |

* **Historial de solicitudes:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Descripción** | **Tipo** | **Ataque asociado** |
| Num-Req-Same-Host | Número de peticiones emitida por la misma fuente | Entero positivo | Flooding, escaneo de vulnerabilidades |
| Num-Req-Same-URL | Número de peticiones con la misma URL | Entero positivo | Flooding de la misma fuente/múltiples fuentes. |
| Num-Req-Same-Host-Diff-URI | Número de peticiones emitida por la misma fuente y solicitando URLs diferentes | Entero positivo | Escaneo de vulnerabilidades |
| Inter-Req-Interval | Intervalo de tiempo de la solicitud Inter | Entero positivo | Flooding, escaneo de vulnerabilidades |

En [2] el autor lista una serie de características identificadas en una petición HTTP y las subdivide en categorías, de acuerdo al aspecto de la petición:

* **URL**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Navegación forzada | URL ilegal que permita el acceso a la página forzando una URL. |
| Violación de acceso | URL contiene caracteres no permitidos o un tiempo de acceso válido expirado. |
| Hijack a la sesión | URL contiene ID de la sesión inválida La URL contiene un valor de ID de sesión que es igual a la ID de sesión que el servidor estableció para esta sesión. |
| Overflow del buffer | Longitud de la URL inaceptable |

* **Método**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Pérdida de información | Utilizar métodos HTTP ilegales en la petición o métodos SOAP no permitidos. |
| Violación RFC | GET o HEAD contienen body. |
| Overflow del buffer | Longitud de la petición POST ilegal. |
| Violación de entrada | La petición contiene string de consulta o datos POST no permitidos. |
| HTTP Request Smuggling | Longitud de cero en el contenido POST |

* **Host**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Cliente no navegador | Cabecera de host no existe en las peticiones HTTP 1.1. Cabecera de host contiene dirección IP. |

* **Parámetros**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Violación de acceso | Los parámetros de la petición contienen caracteres ilegales. |
| Manipulación de parámetros | La solicitud contiene un tipo de datos de parámetro ilegal, contiene un valor numérico de parámetro que no está en el rango permitido, contiene un valor de parámetro estático ilegal o contiene un valor de parámetro alfanumérico que no cumple con el campo de expresión regular. |
| Violación de entrada | La solicitud contiene un parámetro dinámico cuyo valor cambió a vacío o el valor contiene un meta carácter ilegal, contiene un número ilegal de parámetros obligatorios o puede contener un parámetro no permitido. La longitud del valor del parámetro no está permitida o la solicitud de varias partes tiene un parámetro con un valor NULL. |
| HTTP parameter pollution attack | La petición contiene el mismo nombre que muchos nombres de parámetros o contien decodificación múltiple para el URI o el valor del parámetro. |

* **Cookies**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| HTTP parser attack | La header del cookie no es compatible con RFC. |
| Violación de acceso | Cookie de sesión CSRF inyectada en la respuesta. |
| Overflow del buffer | La petición contiene una longitud de la cookie ilegal. |
| Violación de cookie | La petición tiene un dominio de cookie modificado no permitido. |

* **Referrer URL**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Ataque automatizado | El header del referrer contiene una referrer URL no identificada. |

* **Información delicada**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Pérdida de información | La respuesta contiene información delicada como información de tarjetas para los pagos. |

* **Archivo sospechoso**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Virus | La petición incluye un archivo que contiene un gusano o un virus. |

* **Patrón sospechoso**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Firma de ataque | La petición o la respuesta contiene un patrón que concuerda con la firma de un ataque. |

* **IP**

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Ataque DoS | La petición contiene una IP desconocida o que esté en la lista negra. |

**Para XSS**

En [3] se propone 70 características de detección para ataques XSS obtenidas mediante un

método de extracción de características. El autor no especifica la descripción de cada característica pero el nombre es bastante descriptivo:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Number\_Of\_Redirections | Difference\_In\_Redirections | Number\_Of\_Instantiated\_Objects | Difference\_In\_Instantiated\_Objects | Lenght\_Of\_Longest\_Single\_Evaluated\_Code |
| Total\_Bytes\_Of\_Evaluated\_Code | Total\_Bytes\_Allocated | Number\_Of\_Executions | Length\_Of\_Longest\_Unprintable\_String | Number\_Of\_Unprintable\_Strings |
| Ratio\_Of\_Definitions\_To\_Uses | Lenght\_Of\_Method\_Call\_Parameter | Total\_Number\_Of\_Method\_Calls | Memory\_Overflow\_Ocurred | Unprintable\_String\_Used\_For\_Instantiated\_Oject\_Parameter |
| Unprintable\_Strings\_AND/OR\_Redirects | Method\_Cal | open | run | Savetofile |
| Send | Setattribute | Write | timespanformat | shellexecute |
| playerproperty | uploadlogs | hgs\_startnotify | downloadandinstall | getvariable |
| linksbicons | openurl | getresponseheader | keyframe | setrequestheader |
| Setslice | buildpath | getspecialfolder | environment | rawparse |
| iestartnative | setformatlikesample | createobject | specialfolders | replace |
| createnewfoldername | addevent | isversionsupported | split | new |
| evaluate | msdatasourceobject | allowscriptaccess | concat | url |
| mode | type | Import | close | allowcontextmenu |
| cachefolder | compressedpath | console | printsnapshot | shownavigationbuttons |
| snapshotpath | wkspictureinterface | zoom | script\_Size | intent |

Como se puede observar, los autores proponen clasificaciones diferentes para las características pero estas no difieren mucho de una investigación a otra; en todas se tiene en cuenta el análisis de características muy comunes en todos los ataques como lo son el análisis de las cabeceras de la petición, al igual que el llamado a métodos o funciones JavaScript (en el caso preciso de los ataques XSS) y el ingreso de sentencias SQL (para el caso de SQL Injection).

Las características antes mencionadas servirán de apoyo para seleccionar las características que mejor se ajustan a cada tipo de ataque que se va a tratar. Cabe resaltar que para el fin de este proyecto, se debe hacer una depuración de estas características, quedándose únicamente con aquellas que en mayor medida puedan identificar correctamente los ataques a trabajar.

**Bibliografía**

[1] B. Salem and T. Karim, “Classification features for detecting server-side and client-side web attacks,” in IFIP International Federation for Information Processing, 2008, vol. 278, pp. 729–733.

[2] M. A. Wazzan and M. H. Awadh, “Towards improving web attack detection: Highlighting the significant factors,” in 2015 5th International Conference on IT Convergence and Security, ICITCS 2015 - Proceedings, 2015.

[3] N. Khan, J. Abdullah, and A. S. Khan, “Defending malicious script attacks using machine learning classifiers,” Wirel. Commun. Mob. Comput., vol. 2017, pp. 1–9, 2017.