**Analytics Use Cases**

**Detecting the following strings in the IIS logs**

**XSS Detection**

* <script>alert(xss)</script>
* ”><script>alert(document.cookie)</script>
* aaaa”><script>alert(1)</script>
* <script>prompt(’1’)</script>
* ‘><script>alert(document.cookie)</script>
* <script>alert(‘xss’);</script>
* <scr<script>ipt>alert(xss)</scr</script>ipt>
* <script><script>alert(1)</script>
* <script language=”javascript”>window.location.href = ”beeftrap.html” ; </script>
* <script src=”<http://beefhook.js>”></script>
* <ScRiPt>alert(1)</ScRiPt>
* %00<script>alert(1)</script>
* <img onerror=alert(1) src=a>

**SQL Injection Detection –**

* 1 OR 1=1
* ’ OR 1=1 --
* ” OR 1=1 --’
* OR 1=1;
* 1 AND 1=1
* x’ OR ’1’=’1
* ‘ OR 1 in (@@version)--
* ‘ UNION (select @@version) --
* 1 OR sleep(\_\_\_TIME\_\_\_)#
* ’ OR sleep(\_\_\_TIME\_\_\_)#
* ” OR sleep(\_\_\_TIME\_\_\_)#
* 1 or benchmark(10000000,MD5(1))#
* ’ or benchmark(10000000,MD5(1))#
* ” or benchmark(10000000,MD5(1))#
* ;waitfor delay '0:0:\_\_TIME\_\_'--
* );waitfor delay '0:0:\_\_TIME\_\_'--
* ';waitfor delay '0:0:\_\_TIME\_\_'--
* ";waitfor delay '0:0:\_\_TIME\_\_'--
* OR 1=1 ORDER BY table\_name DESC
* x’; UPDATE table SET value WHERE user=’x
* 1’; INSERT INTO table VALUES(‘value’,‘value’);--
* 101 AND (SELECT ASCII(SUBSTR(name,1,1)) FROM table WHERE foo=n)$ --
* ’ union select null,LOAD\_FILE(’../../../../../etc/passwd’),null,null,null –

**Commands to detect Web attacks**

**Regex to identify attempts to exploit Stored Procedures** - /exec(\s|\+)+(s|x)p\w+/ix

**Regex to identify “|” symbol** - /(\||%00|system\(|eval\(|`|\\)/i

**Regex to identify Remote File Inclusion -** /(https?|ftp|php|data):/i

**Regex to identify Directory\Path Traversal or dot-dot-slash attack -** /(\.|(%|%25)2E)(\.|(%|%25)2E)(\/|(%|%25)2F|\\|(%|%25)5C)/i

**Regex to identify Session IDs, user/session information, user names, passwords –**

/ \.jsp.\*\?.\*(userId|password)=./

/;jsessionid=./

**Detecting Shell executions**

`shell\_command` - executes the command

$(shell\_command) - executes the command

| shell\_command - executes the command and returns the output of the command

|| shell\_command - executes the command and returns the output of the command

; shell\_command - executes the command and returns the output of the command

&& shell\_command executes the command and returns the output of the command

> target\_file - overwrites the target file with the output of the previous command

>> target\_file - appends the target file with the output of the previous command

< target\_file - send contents of target\_file to the previous command

- operator - Add additional operations to target command

**Directory\Path Traversal detection**

* etc/passwd
* /etc/passwd%00
* ../etc/passwd
* ../../etc/passwd
* ../../../etc/passwd
* ../../../../etc/passwd
* ../../../../boot/grub/grub.conf
* ../../../../../var/log
* ../../../../../etc/apache2/httpd.conf
* ..\..\..\../c/boot.ini
* ..\../..\../boot.ini
* ../../../../../../etc/shadow&=%3C%3C%3C%3C%3C
* ..%2F..%2F..%2F..%2F..%2F..%2Fetc%2Fpasswd
* %2E%2E%2F%2E%2E%2F%2E%2E%2Fetc%2Fpasswd
* ..%5c..%5c..%5c..%5c..%5c..%5cc/boot.ini
* /%c0%ae%c0%ae/%c0%ae%c0%ae/%c0%ae%c0%ae/etc/passwd
* /admin/index.html
* //admin/index.html
* /admin/./index.html
* /admin/../admin/index.html
* /admin/./../admin/index.html

**Double Encoding**

%253Cscript%253Ealert('XSS')%253C%252Fscript%253E

**Regex to Identify – Base64 Strings**

“\A[a-zA-Z\d\/+]+={,2}\z”

“/^(?:[A-Z0-9+\/]{4})\*(?:[A-Z0-9+\/]{2}==|[A-Z0-9+\/]{3}=|[A-Z0-9+\/]{4})$/i;”

“^[A-Za-z0-9+\/=]+\Z”

**SQL Injection Attack Patterns Detection**

Regex to Identify SQL Keywords - /((\%27)|(\'))(select|union|insert|update|delete|replace|truncate)/ix

Regex to identify ‘ & -- delimiter - /(\')|(\%27)|(\\)|(#)|(\%23)/ix

Regex to identify “=” - /((\%3D)|(=))[^\n]\*((\%27)|(\')|(\-\-)|(\%3B)|(;))/i

Regex to identify attempts to exploit Stored Procedures - /exec(\s|\+)+(s|x)p\w+/ix

Regex to identify “|” symbol - /(\||%00|system\(|eval\(|`|\\)/i

Regex to identify Remote File Inclusion - /(https?|ftp|php|data):/i

Directory\Path Traversal Attack Patterns

Regex to identify Directory\Path Traversal or dot-dot-slash attack - /(\.|(%|%25)2E)(\.|(%|%25)2E)(\/|(%|%25)2F|\\|(%|%25)5C)/i

==

**Malicious Filename and paths**

Mimikatz.exe

mimilove.exe

mimidrv.sys

mimilib.dll

SEKURLSA

LSADUMP

\.kirbi$

c:\temp\lsass.dmp

lsass.dmp

c:\temp\mimikatz

./mimikatz

Invoke-Mimikatz.ps1

Invoke-mimikittenz.ps1

**Clear Text Credit cards detection**

"/5\d{3}(\s|-)?\d{4}(\s|-)?\d{4}(\s|-)?\d{4}/" MasterCard number

"/4\d{3}(\s|-)?\d{4}(\s|-)?\d{4}(\s|-)?\d{4}/" Visa card number

"/3\d{3}(\s|-)?\d{6}(\s|-)?\d{5}/" American Express card number

==

**Regex to detect the malicious sites**

/wp-(includes|admin|content)/.\*\.(exe|dll|scr)

**Regex to detect the Onion and Tor sites**

[a-z0–9]{16}\.(onion|tor2web|torlink)\.