YARA

cyberdefenses.com

Rule Marking

rules can be marked global or private

private rule privaterule

global rule globalrule

- private rules are not reported by YARA when they match
- global rules are applied to all rules at once and evaluated before the rest of the rules

```
rule  Rule_name : tag1 tag2
 {
```

meta:

*****None of these are required*****

description = "description"

author = "author"

reference = "reference"

date = "date"

usage = "usage guidelines"

include = "include directive file"

strings:

$text = "text here" nocase wide ascii fullword

****

nocase = case insensitive

wide = Unicode or 2 bytes per char

ascii = use with wide to search ascii as well

fullword = match only if delimited/not partials

****

$hex = {CA FE BE [1-4] ?? ?? (16 13 | 33 41) BE}

****

? = wild card

[#-#] = arbitrary bytes

(a | b) = (a or b)

****

$regex = /[029a-fA-F]{32}/

() = grouping

[] = character class

{a} = match exactly *a* times

{a,} = match at least *a* times

{,b} = match 0 to *b* times

{a,b} = match *a* to *b* times

* = match 0 or more times

+ = match 1 or more times

? = match 0 or 1 times

//add a ? to make greedy expressions repeat as few times as possible, such as .+? Or {3,6}?

\ = escape the next metacharacter

by Monty St John

^ = match the beginning

$ = match the ending

| = alternation

\t = tab

\n = new line

\w = word

\s = whitespace

\s = decimal digit

\D = non-digit

condition:

$text or ($hex and not $regex)

****

Boolean *and, or and not*

Relational operators *>=, <=, >, <, ==* and *!=*

Arithmetic operators +, -, \, *, %

Bitwise operates *&, |, <<, >>, !,* and *^*

Counting

#text == 6 and #hex > 5

****x offset or in range a..z****

$hex at 250 or $hex in (0..filesize)

****filesize special variable****

filesize < 3000KB

****entrypoint special variable****

$hex in (entrypoint..entrypoint + 10)

Sets of strings

all of them = all strings in the rule

any of them = any string in the rule

all of ($a*) = all strings those identifier starts with "$a"

any of ($a, $b, $c) = any of $a, $b or $c

1 of ($*) = same as "any of them"

X of them = matches any x of the strings

Iteration of strings

for any of ($a, $b, $c) = any of $a, $b or $c applies the expression to multiple strings

for all | in (1,2,3) : (@a[i] + 10 == @b[i]) = first three occurrences of $b should be 10 bytes away from the first three occurrences of $a

Reference another rule

```
$text and my_other_rule
{
```