Decoder matching with hostname

234 views

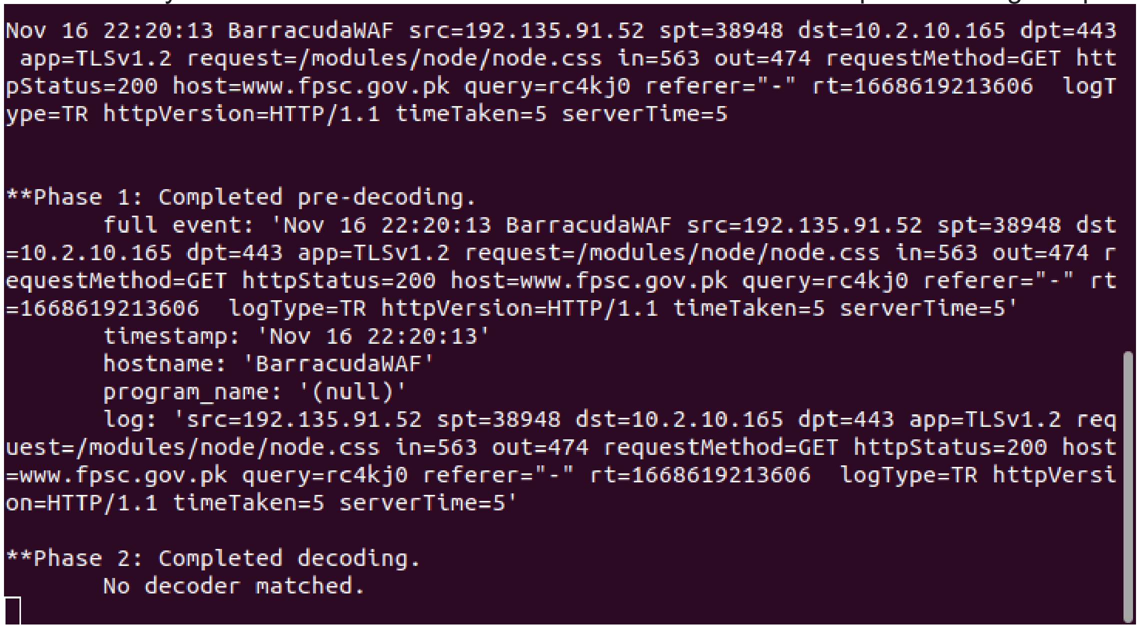
Subscribe

Usama Rajput

to Wazuh mailing list

Nov 21, 2022, 1:16:18 AM

Is there a way I can match decoder with the hostname because after pre-decoding I'm practically left with nothing in the log to match with...



```
Nov 16 22:20:13 BarracudaWAF src=192.135.91.52 spt=38948 dst=10.2.10.165 dpt=443
 app=TLSv1.2 request=/modules/node/node.css in=563 out=474 requestMethod=GET htt
pStatus=200 host=www.fpsc.gov.pk query=rc4kj0 referer="-" rt=1668619213606  logT
ype=TR httpVersion=HTTP/1.1 timeTaken=5 serverTime=5


**Phase 1: Completed pre-decoding.
       full event: 'Nov 16 22:20:13 BarracudaWAF src=192.135.91.52 spt=38948 dst
=10.2.10.165 dpt=443 app=TLSv1.2 request=/modules/node/node.css in=563 out=474 r
equestMethod=GET httpStatus=200 host=www.fpsc.gov.pk query=rc4kj0 referer="-" rt
=1668619213606  logType=TR httpVersion=HTTP/1.1 timeTaken=5 serverTime=5'
       timestamp: 'Nov 16 22:20:13'
       hostname: 'BarracudaWAF'
       program_name: '(null)'
       log: 'src=192.135.91.52 spt=38948 dst=10.2.10.165 dpt=443 app=TLSv1.2 req
uest=/modules/node/node.css in=563 out=474 requestMethod=GET httpStatus=200 host
=www.fpsc.gov.pk query=rc4kj0 referer="-" rt=1668619213606  logType=TR httpVersi
on=HTTP/1.1 timeTaken=5 serverTime=5'

**Phase 2: Completed decoding.
       No decoder matched.
```

Jonathan Martín Valera

to Wazuh mailing list

Nov 21, 2022, 3:14:12 AM

Hi,

No, it is not possible to link the pre-decoded fields with the decoded ones, but in this case, **it is not necessary** to generate the alerts you want. Let me explain with some examples.

Imagine that we want to generate a generic alert whenever we receive a log like that and the pre-decoded hostname has the value `BarracudaWAF`. For this, it would not be necessary to create any decoder, since with the pre-decoded values we can identify the log and generate the corresponding alert. An example of such a rule would be the following:

```
<rule id="100150" level="3">
    <hostname>BarracudaWAF</hostname>
    <description>Test rule</description>
</rule>
```

If we test the log in the `wazuh-logtest` tool, we can see how it would generate alert only with that rule:

```
Nov 16 22:20:13 BarracudaWAF src=192.135.91.52 spt=38948 dst=10.2.10.165 dpt=443 app=TLSv1.2 request=/modules/node/node.css in=563

**Phase 1: Completed pre-decoding.
    full event: 'Nov 16 22:20:13 BarracudaWAF src=192.135.91.52 spt=38948 dst=10.2.10.165 dpt=443 app=TLSv1.2 request=/modules/node
    timestamp: 'Nov 16 22:20:13'
    hostname: 'BarracudaWAF'

**Phase 2: Completed decoding.
    No decoder matched.

**Phase 3: Completed filtering rules.
    id: '100150'
    level: '3'
    description: 'Test rule!'
    groups: '['custom_rule']'
    firedtimes: '1'
    mail: 'False'
**Alert to be generated.
```

Now, imagine that you want a more specific condition in order to generate the alert, for example, the hostname must be `BarracudaWAF` and the `httpStatus` field must be 200:

```
<rule id="100150" level="3">
    <hostname>BarracudaWAF</hostname>
    <regex>httpStatus=200</regex>
    <description>Test rule</description>
</rule>
```

If we test the log again:

```
Nov 16 22:20:13 BarracudaWAF src=192.135.91.52 spt=38948 dst=10.2.10.165 dpt=443 app=TLSv1.2 request=/modules/node/node.css in=563

**Phase 1: Completed pre-decoding.
```

```
    full event: 'Nov 16 22:20:13 BarracudaWAF src=192.135.91.52 spt=38948 dst=10.2.10.165 dpt=443 app=TLSv1.2 request=/modules/node
    timestamp: 'Nov 16 22:20:13'
    hostname: 'BarracudaWAF'

**Phase 2: Completed decoding.
    No decoder matched.

**Phase 3: Completed filtering (rules).
    id: '100150'
    level: '3'
    description: 'Test rule'
    groups: '['custom_rule']'
    firedtimes: '1'
    mail: 'False'
**Alert to be generated.
```

On the other hand, if you want to make use of decoders and decoded fields, then what we can do is to use the log structure itself beyond the pre-decoded to identify it. For example, in this case, I will use the fields themselves and the order to identify them (look at the `<prematch>` of the following decoder). The decoder would look like this:

```
<decoder name="test">
    <prematch>src=\.* spt=\.* dst=\.* dpt=\.* app=\.* request=\.* in=\.* out=\.* requestMethod=\.* httpStatus=\.* host=\.* query=\.
    <regex>src=(\.*) spt=(\.*) dst=(\.*) dpt=(\.*) app=(\.*) request=(\.*) in=(\.*) out=(\.*) requestMethod=(\.*) httpStatus=(\.*)
    <order>src, spt, dst, dpt, app, request, in, out, requestMethod, httpStatus, host, query, referer, rt, logType, httpVersion, ti
</decoder>
```

Regarding the rule, we can now make use of the created decoder name and decoded fields to make our own regex:

```
<rule id="100150" level="3">
    <decoded_as>test</decoded_as>
    <hostname>BarracudaWAF</hostname>
    <field name="httpStatus">200</field>
    <description>Test rule!</description>
</rule>
```

If we test the log again:

```
**Phase 1: Completed pre-decoding.
    full event: 'Nov 16 22:20:13 BarracudaWAF src=192.135.91.52 spt=38948 dst=10.2.10.165 dpt=443 app=TLSv1.2 request=/modules/node
    timestamp: 'Nov 16 22:20:13'
    hostname: 'BarracudaWAF'

**Phase 2: Completed decoding.
    name: 'test'
    app: 'TLSv1.2'
    dpt: '443'
    dst: '10.2.10.165'
    host: 'www.fpsc.gov.pk'
    httpStatus: '200'
    httpVersion: 'HTTP/1.1'
    in: '563'
    logType: 'TR'
    out: '474'
    query: 'rc4kj0'
    referer: '"-"'
    request: '/modules/node/node.css'
    requestMethod: 'GET'
    rt: '1668619213606'
    serverTime: '5'
    spt: '38948'
    src: '192.135.91.52'
    timeTaken: '5'

**Phase 3: Completed filtering (rules).
    id: '100150'
    level: '3'
    description: 'Test rule!'
    groups: '['custom_rule']'
    firedtimes: '1'
    mail: 'False'
**Alert to be generated.
```

As you can see, there are many ways to achieve the expected result.

I hope it has helped you. Try it and let us know.

Best regards.