Suricata - Feature #1872

add --list-decoder-protos or similar

08/26/2016 09:42 AM - Peter Manev

Status: Assigned
Priority: Normal
Assignee: Andreas Herz
Category:
Target version: TBD
Effort:
Difficulty:

Description
It would be useful to have

```
suricata --list-decoder-protos
```

or similar to list supported decoder protocols like we have:

```
pevma@DONPEDRO:~$ sudo suricata --list-app-layer-protos
========Supported App Layer Protocols========
http
ftp
smtp
tls
ssh
imap
msn
smb
dcerpc
dns
```

Related issues:
Related to Bug #635: Some keywords missing in list-keyword command

History
#1 - 09/06/2016 04:03 PM - Andreas Herz
- Assignee set to OISF Dev
- Target version set to TBD

#2 - 05/31/2019 09:55 PM - Andreas Herz
- Related to Bug #635: Some keywords missing in list-keyword command added

#3 - 06/06/2019 11:42 AM - Victor Julien
- Status changed from New to Assigned
- Assignee changed from OISF Dev to Andreas Herz

#4 - 06/14/2019 10:02 PM - Andreas Herz
While the app-layer-protocols are also keywords usable in rules not all decode protos are real keywords (vlan, pppoe f or example), so should we still print it the same way?

#5 - 06/15/2019 07:09 AM - Peter Manev
Maybe have a message per field that is not a keyword? could be messy though.

#6 - 06/20/2019 08:21 PM - Victor Julien
I think these are different things. We have protocols that suri can decode and protocol names for in rules. I don't mind having 2 options to list each
Just to be sure, you would suggest to split those into two options like --list-decoder-protos and --list-decoder-protos-keywords (names still to be discussed)?

Yeah. I would think --list-decoder-protos and --list-rule-protos

I like that approach.

I can implement that but --list-decoder-protos would still have all and --list-rule-protos would be a subset excluding those which aren't keywords. But while playing around with #635 I would either add those of the --list-rule-protos to the --list-keywords list (to match idea 1) or as a section (to match idea 2).

I'm confused with what you're asking/saying, but I think its best to start with an implementation and then we can discuss the result/output. It's not a big project so it won't be a waste of time if things need to change.