API CSV data Export for Grafana

currently (Grafana version 7) it seems to not be possible to create a useful CSV table directly out of a Grafana Panel via the API. It is possible on the GUI via the “Inspect” functions, but it still seems to be unavailable through the API.

looking at the requests sent to the grafana server during the Inspect action, i found a query like this one:

```
http://myserver:3000/api/datasources/proxy/1/render?target=powermeter.phase*
.watt&from=1598939104&until=1598939114&format=json
```

which provides the entire dataset for the given targets for the given time frame in json. You can actually change the `format=json` to `format=csv` but the resulting csv is garbage:

```
powermeter.phase1.watt,2020-09-01 05:45:05,707.3
powermeter.phase1.watt,2020-09-01 05:45:06,707.1
powermeter.phase1.watt,2020-09-01 05:45:07,707.8
powermeter.phase1.watt,2020-09-01 05:45:08,715.0
powermeter.phase1.watt,2020-09-01 05:45:09,707.8
powermeter.phase1.watt,2020-09-01 05:45:10,707.9
powermeter.phase1.watt,2020-09-01 05:45:11,707.4
powermeter.phase1.watt,2020-09-01 05:45:12,707.5
powermeter.phase1.watt,2020-09-01 05:45:13,706.8
powermeter.phase1.watt,2020-09-01 05:45:14,707.3
powermeter.phase2.watt,2020-09-01 05:45:05,740.7
powermeter.phase2.watt,2020-09-01 05:45:06,741.9
powermeter.phase2.watt,2020-09-01 05:45:07,742.8
powermeter.phase2.watt,2020-09-01 05:45:08,739.4
powermeter.phase2.watt,2020-09-01 05:45:09,743.7
powermeter.phase2.watt,2020-09-01 05:45:10,741.8
powermeter.phase2.watt,2020-09-01 05:45:11,740.7
powermeter.phase2.watt,2020-09-01 05:45:12,741.2
powermeter.phase2.watt,2020-09-01 05:45:13,741.7
powermeter.phase2.watt,2020-09-01 05:45:14,741.0
powermeter.phase3.watt,2020-09-01 05:45:05,-0.0
powermeter.phase3.watt,2020-09-01 05:45:06,0.0
powermeter.phase3.watt,2020-09-01 05:45:07,0.2
powermeter.phase3.watt,2020-09-01 05:45:08,0.0
powermeter.phase3.watt,2020-09-01 05:45:09,0.1
powermeter.phase3.watt,2020-09-01 05:45:10,0.0
powermeter.phase3.watt,2020-09-01 05:45:11,0.1
powermeter.phase3.watt,2020-09-01 05:45:12,0.1
powermeter.phase3.watt,2020-09-01 05:45:13,0.0
powermeter.phase3.watt,2020-09-01 05:45:14,0.0
```

while this is technically speaking csv, it is of little use if you want to import the dataset into excel for visualisation or any other purposes.

so i wrote a little PHP script which i am hosting on the grafana server itself and which allows me to get a nice table with a separate column for each of the requested targets like so:

```
curl -s
"http://myserver/csv.php?target=powermeter.phase*.watt&from=1598939104&until
=1598939114"
time,powermeter.phase1.watt,powermeter.phase2.watt,powermeter.phase3.watt
2020-09-01T07:45:05+02:00,707.3,740.7,-0
2020-09-01T07:45:06+02:00,707.1,741.9,0
2020-09-01T07:45:07+02:00,707.8,742.8,0.2
2020-09-01T07:45:08+02:00,715,739.4,0
2020-09-01T07:45:09+02:00,707.8,743.7,0.1
2020-09-01T07:45:10+02:00,707.9,741.8,0
2020-09-01T07:45:11+02:00,707.4,740.7,0.1
2020-09-01T07:45:12+02:00,707.5,741.2,0.1
2020-09-01T07:45:13+02:00,706.8,741.7,0
2020-09-01T07:45:14+02:00,707.3,741,0
```

to specify multiple targets the `target`parameter has to be formatted as an array:

```
curl -s
"http://myserver/csv.php?target[]=time,powermeter.phase1.watt&target[]=power
meter.phase2.watt&target[]=powermeter.phase3.watt&from=1598939104&until=1598
939114"
```

would produce the same output as the above query

this makes it much easier for me to process the data afterwards in libre office calc or excel or similar tools.

here is the quick and dirty php script that i put into the document root of my grafana host, serving it via apache.

please note: *i have saved the API key directly into my script, as this server is not publicly accessible and holds no confidential data anyways.. however, consider moving the API key out of the script if you use id on a public server*

csv.php

```
<?php
$SEPARATOR=",";
$APIKEY="<your key goes here>";
$HOST="myserver:3000";
$PROT="http";

#$queryTargets=array("powermeter.phase1.watt","sdr.cmc01.SYS_POWER","po
wermeter.phase2.watt");
$queryTargets=$_REQUEST["target"];
if (!is_array($queryTargets)){
    $queryTargets=array($_REQUEST["target"]);
}
```

```
if (count($queryTargets)>0){
    $queryTargetsString="target=";
    foreach ($queryTargets as $qt){
        $queryTargetsString.=urlencode($qt)."&target=";
    }
    $queryTargetsString=substr($queryTargetsString,0,-8);
} else {
    throw new Exception("no targets defined, specify at least one
target");
}
$from=(int)$_REQUEST["from"];
$until=(int)$_REQUEST["until"];

if ($from <= 0 || $until <=0 ){
    throw new Exception("no fram and u ntil parameters defined");
}
header('Content-type: text/csv');
header('Content-disposition: attachment;filename=data.csv');

$json = file_get_contents("$PROT://api_key:$APIKEY@".
                          "$HOST/api/datasources/proxy/1/render?".
                          $queryTargetsString.
                          "&from=$from&until=$until".
                          "&format=json");
$data = json_decode($json,true);
if (!$data){
    die();
}
$table=array();
$targets=array();
if(count($data) > 0 ){
    foreach ($data as $dataset){
        $target=$dataset["target"];
        $targets[]=$target;
        foreach ($dataset["datapoints"] as $point){
            $table[$point[1]][$target]=$point[0];
        }

    }

    echo "time$SEPARATOR".implode($SEPARATOR,$targets)."\n";
    foreach ($table as $date=>$line){
        echo date("c",$date);
        foreach($targets as $target){
            echo $SEPARATOR.$line[$target];
        }
        echo "\n";
    }
}
?>
```

From:
http://wiki.psuter.ch/ - **pswiki**

Permanent link:
http://wiki.psuter.ch/doku.php?id=api_csv_export_for_grafana&rev=1598968917

Last update: **01.09.2020 16:01**