## Geolocation and other services (netstat, DNS, top, time, GeoIP)

The commands used in this manual assume that your hosting is based on a UNIX/Linux server. If it is on MS Windows Server, some commands should be replaced with their equivalents: traceroute should be replaced with the tracert command, etc. Note: some commands on your hosting have perhaps been deliberately blocked for security reasons by the administrator of your hosting. When testing the operation of external portals using proposed scripts, you will find that often the geolocation information is not correct, to ensure customer privacy - that is, to protect their physical location when they enter portals that register geolocation data.

When you create the computer measurement system to monitor any process - you have to be sure it is working all the time. Periodical monitoring services running on a given device (Arduino with Ethernet Shield, Raspberry Pi or just PC server) - TCP and UDP ports on which it listens, active TCP network connections and many other tests can be implemented from other computers or microcontroller in the IP network or from the same computer or microcontroller that is the server of these services. There are many applications that allow for monitoring services in IP networks, e.g. Nagios https://en.wikipedia.org/wiki/Nagios

Before you start creating your own application of this type, it is worth reviewing the article on the comparison of the functionality of different network monitoring systems: https://en.wikipedia.org/wiki/Comparison_of_network_monitoring_systems.

Building an application with Nagios functionality would be time-consuming task, therefore during laboratory classes it is possible only to test codes that implement small sections of tasks related to the above-mentioned topics. At the first beginning you should test the *netstat* command, which can be called from the command line on UNIX and Windows systems.

## Implementation of selected tasks during classes

1. For the exercise nr 2 please create a subfolder /z2 in your /public_html folder in your hosting.
2. index.php file located in that /z2 folder should be traditionally the start file for that exercise.
3. To present the operation of the tested scripts on your hosting please create the next files index1.php, index2.php etc., then put the links to those files in index.php file - the main file of this exercise. Thanks to this, the individual steps of the task can be independently verified by running links that call them.
4. Please read the description of the command netstat https://en.wikipedia.org/wiki/Netstat
5. Please run the *cmd* command in your Windows system and run the command netstat. Please analyze the displayed data.



6. You can find the description of the netstat.php script here: https://github.com/netstat-php/netstat_php, it works as the simplified version of the classical netstat application. Please copy the whole script that is shared in this link and test it in your hosting.
   https://fam.tuwien.ac.at/~schamane/_/netstat_php.
   https://fam.tuwien.ac.at/~schamane/sysadmin/netstat/netstat.phps
   - Please place the link index1.php to that script in your main file index.php of that exercise.
   - Please extend this script to detect more services working on your hosting
     - o there are services that work on your hosting – you can be sure that they work, because you use them – webserver on port 80 for instance,
     - o consider why the operation of some services may be hidden on the hosting server,
     - o check which services are not visible with this checking method.

7. Read the description of the php.info function at http://php.net/manual/en/function.phpinfo.php.
Test the phpinfo () script on your hosting, after reviewing its report, list the ports that can be said to be open:

```php
<?php
    echo phpinfo ();
?>
```

8. If you have additional privileges on your hosting server – you can use exec command, that allows to run console commands. Verify that the hosting server identifies you as a specific user, when you work on it, after you have logged in to your hosting, check if the same username will be displayed if you enter your URL on a different computer.
Note: on many web hosts this script will not display the correctly identified name of the hosting users, the effect of this script is related to the administrative settings and in this case.
If the script does not work correctly – do not worry.

```php
<?php
    echo exec ('whoami');
?>
```

9. Test the following scripts concerning DNS. Check the functions used in these scripts to use them correctly in your own applications:
http://php.net/manual/en/function.dns-get-record.php
http://php.net/manual/en/function.gethostbyname.php
http://php.net/manual/en/reserved.variables.server.php
http://php.net/manual/en/function.gethostbyaddr.php

```php
<?php
        $result = dns_get_record("fb.com");
        print_r($result);
?>
```

```php
<?php
        $ip = gethostbyname(' fb.com');
        echo $ip . '<BR />';

        $ip = $_SERVER["REMOTE_ADDR"];
        echo $ip. '<BR />';

        $hostname = gethostbyaddr("8.8.8.8");
        echo $hostname. '<BR />';

        $hostname = gethostbyaddr($_SERVER['REMOTE_ADDR']);
        echo $hostname;
?>
```

10. Test the script displaying the date and time https://en.wikipedia.org/wiki/Unix_time
Please note that Unix time is very convenient for date operations such as calculating the time difference.
This feature can be used widely in created applications, at least until 2038

```php
<?php
        $time1 = time ();
        $time2 = date ("r", $time1);
        echo 'Unix time:   ' . $time1 . "<BR />";
        echo 'Easy to interpret time: ' . $time2;
?>
```

11. Geolocation - this technology uses different mechanisms - local or centralized databases, hence the effects of its operation may be different, sometimes incorrect, and errors may reach many kilometers:
    A. Test web applications that display nodes on the Internet, through which data pass during the test, note that sometimes error data is displayed.
    B. Perform another test with the same tool after 10 seconds, note that sometimes the path through which the data passes completely changes. You can use the tests for URL of your hosting, by the way you can find out more information about your hosting service provider.
       http://www.yougetsignal.com/tools/visual-tracert
       http://www.monitis.com/traceroute
       https://www.ip2location.com/free/traceroute
    C. Verify your current geolocation data, note that full and correct information is not always displayed
       https://ipinfo.io
    D. Please verify other tools:
       http://ipinfo.info/index.php look at the option "My public IP Address": choose more... ->
       http://ipinfo.info/html/privacy-check.php
    E. Please verify other tools:
       https://www.ip2location.com/free.asp
    F. Please run Opera Internet Browser, go to Settings -> Privacy, turn on VPN.
       Please test one of the mentioned above tools to find out geolocation data, please verify what is going on when you turn on VPN in Opera Internet Browser. Please turn off VPN in that browser and verify if you receive the same data which were displayed by other scripts.
    G. Test selected portals using the smartphone browser. Please exchange the information with other students who have smartphones in the same network as yours, on the subject of geolocation regarding their visits to the portal. Verify how the information about geolocation is presented for smartphone owners from other cellular networks.

12. Test the following script to allow the approximate geolocation of guests of your portal and compare the effects of its operation with applications tested earlier.
    • Paste in Google maps the coordinates displayed in the script and verify that they really correspond to the city mentioned in the script.
    • Please note that the script was created based on the JSON format https://en.wikipedia.org/wiki/JSON

```php
<?php
        $ipaddress = $_SERVER["REMOTE_ADDR"];

        function ip_details($ip) {
                $json = file_get_contents ("http://ipinfo.io/{$ip}/geo");
                $details = json_decode ($json);
                return $details;
        }

        $details = ip_details($ipaddress);
        echo $details -> region;          echo '<BR />';
        echo $details -> country;         echo '<BR />';
        echo $details -> city;            echo '<BR />';
        echo $details -> loc;             echo '<BR />';
        echo $details -> ip;              echo '<BR />';
?>
```

13. Use phpMyAdmin to create a new database for your hosting.
    A. In this database, create a new table with the name of the portalguests. In this table you will register the IP address of guests visiting your portal and the time of these visits.
       **portalguests** (**id**, ipaddress, datatime)

| | # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ☐ | 1 | id 🔑 | int(11) | | UNSIGNED | No | *None* | | AUTO_INCREMENT | 🖉 Change | ⊖ Drop | ▼ More |
| ☐ | 2 | **ipaddress** text | | utf8_polish_ci | | No | *None* | | | 🖉 Change | ⊖ Drop | ▼ More |
| ☐ | 3 | **datetime** | datetime | | | No | current_timestamp() | | | 🖉 Change | ⊖ Drop | ▼ More |

datetime – the date and time format to store both things together

Default value – the date and time of creation the datebase record

    B. Create an index.php file which should allow you to save the visitors' IP address to your database and at the same time that script should display this IP address in the table along with the date and time of entry on the website and selected geolocation data.

14. Expand the application according to the lecturer's instructions, each group of students can have a different version of the application to create
    A. Add a column containing a link to Google maps, allowing you to view the map associated with the location data located in the table, and assigned to the record regarding the visitor visiting your portal.
    B. Modify the application, allowing only the portal visitors' data from different IP numbers to be displayed in the table.
    C. Modify the application allowing to display in the table only data about visitors to the portal coming from different IP numbers, while the additional column will display the number of visits to your guest portal from a given IP.