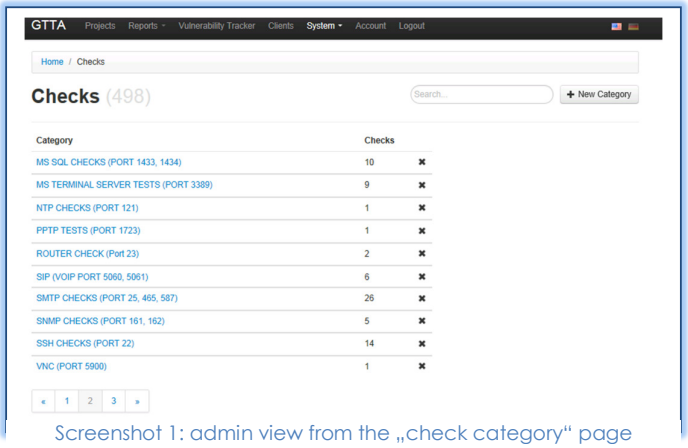
GTTA technical user guide

Creating new checks

Within the category „System Checks“ the admin has the possibility to define the **check categories**. Such categories can be used according to specific test modules like “anonymous web tests” or “VPN checks”. Each check category has **“controls”.** These security controls are safeguards or countermeasures to avoid, counteract or minimize security risks. Security controls can also be categorized according to their nature, for example:

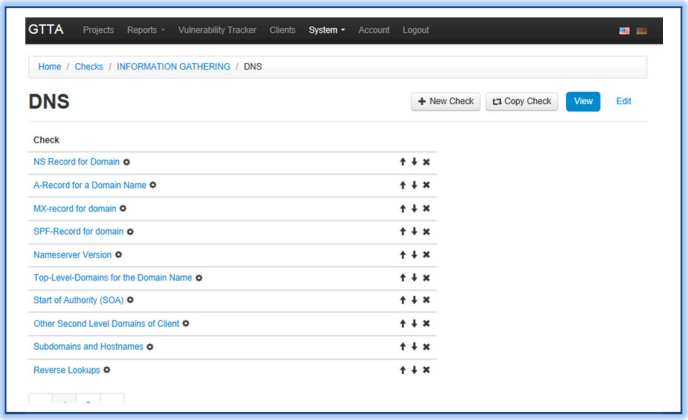
- **Physical controls** e.g. fences, doors, locks and fire extinguishers;
- **Procedural controls** e.g. incident response processes, management oversight, security awareness and training;
- **Technical controls** e.g. user authentication (login) and logical access controls, antivirus software, firewalls;
- **Legal and regulatory or compliance controls** e.g. privacy laws, policies and clauses.

Within each control you will find the actual **checks**. The following screenshot shows the control “DNS” within the category “information gathering”. When you create a new check you can either start from scratch or use an existing one (copy).

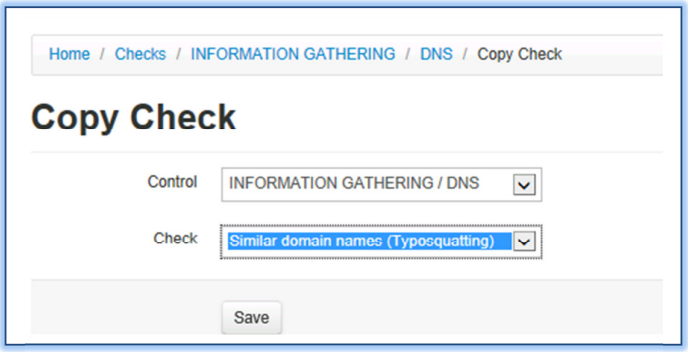


Screenshot 1: admin view from the „check category“ page

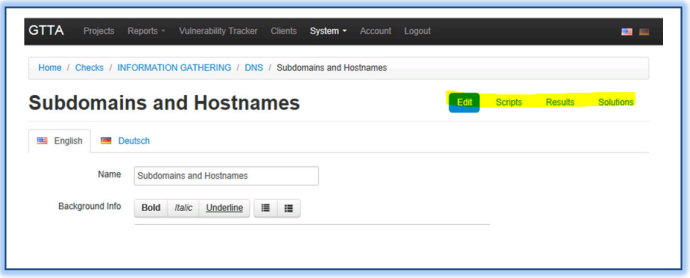
Screenshot 2: admin view of the "control" page that needs to be allocated to the check category



Screenshot 3: admin view of the check page for a specific control



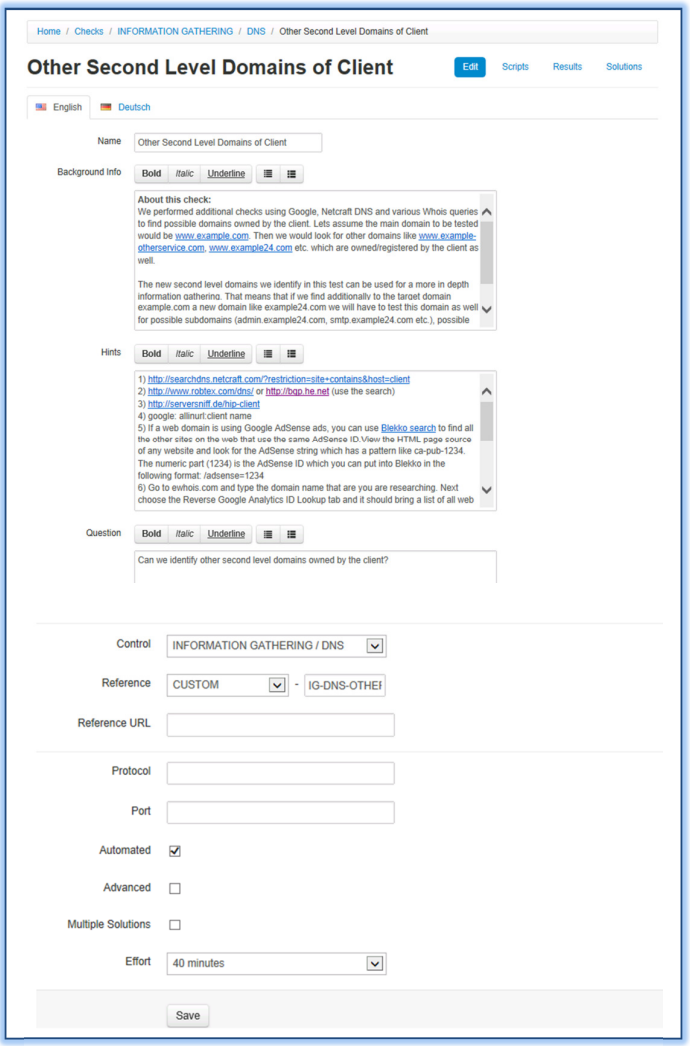
When you edit or add a specific check you will find four menu items:



Screenshot 4: extra options that can be defined for each check

- **Edit**: edit the detailed check info
- **Scripts**: this field is used for automated checks (e.g. if an automated checks requires special input fields from an user). Script integration will be described separately in a later chapter.
- **Results**: here the admin can create predefined results for the test output (e.g. "Good: no vulnerability discovered within test XXX").
- **Solutions**: the admin can add multiple solutions for each issue.

When you open the "**edit check**" page you will be able to customize many things (most of them will carry over to the report). You don't need to use any HTML code for formatting – you can paste the information from other templates directly (e.g. word) and it will keep its formatting.



Screenshot 5: admin view for the detailed check creation page

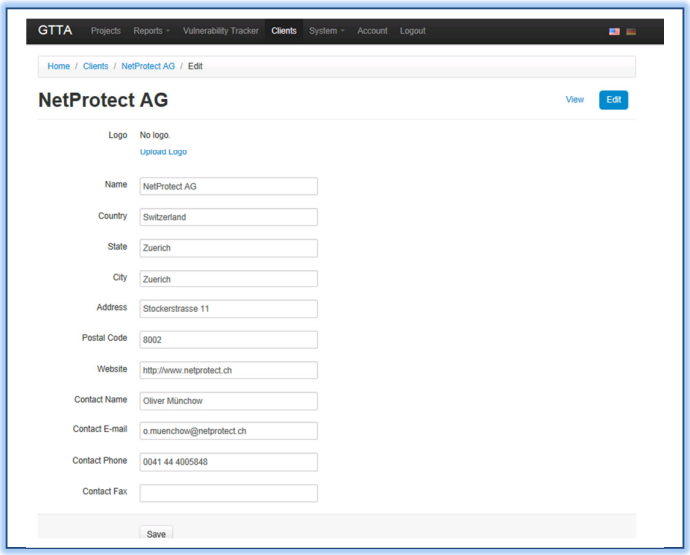
- **Name**: name of the check
- **Background info**: some background info about the check or the topic
- **Hints**: this is a field that only the tester will see. You can provide info's about testing tools, syntax, challenges etc. in here
- **Question**: you define the question, that this check should answer
- **Control:** choose a control (this allows you to move existing checks to another control)
- **Reference**: each check must have an reference (like ISO, OWASP etc.). This reference could be used to select specific test sets in a later stage (e.g. "do only an OWASP TOP 10 Vulnerability test").
- **Script**: if you have an script (perl or

python) you can simply just drag & drop it into a pre-defined directory on the test server and give GTTA its name. With a few minimal changes GTTA will be able to parse the results automatically into the GUI.

- **Effort**: you have to estimate for each test how much time it takes to do the test. This is used in the module "effort estimation"

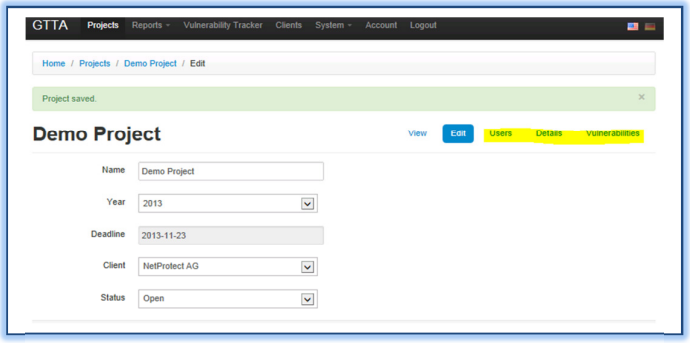
Define new client

Before you can do a test you must first create the client and this is easily done. Just click "clients" from the main menu bar and complete the data. Once saved, all projects allocated to this client will be viewable with one click.



Define new projects

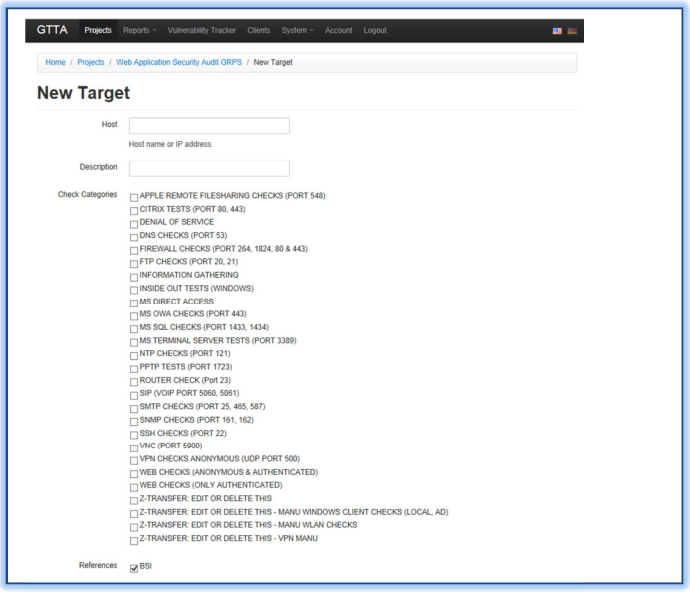
To start a new project, select "projects" and specify the client, start-date and end-date. Once the project is created, a new navigation menu appears to the right:



- **Users:** automatically it will allocate the user who created the project to it. But you can allocate new users with specific right (view, edit etc.) to the project as well.
- **Details:** you can create additional notes with titles for the auditor (e.g. at which business hours the test should take place, who to contact in an emergency etc.)
- **Vulnerabilities:** when clicking on this link it will show you a list of only the vulnerabilities discovered during this test.

Define new targets within a project

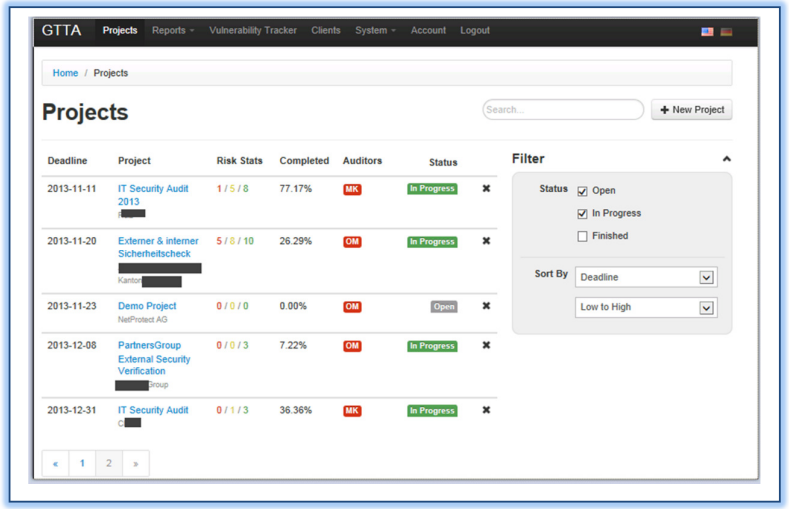
Once you created a project you have the ability to add targets (IP's or hostnames). Next you have to select the type of check category (multiple selections are possible). If you select e.g. just one specific reference type (like OWASP) GTTA will automatically filter all checks and use only the OWASP checks within the test category. For certain tests you can also only use a domain name



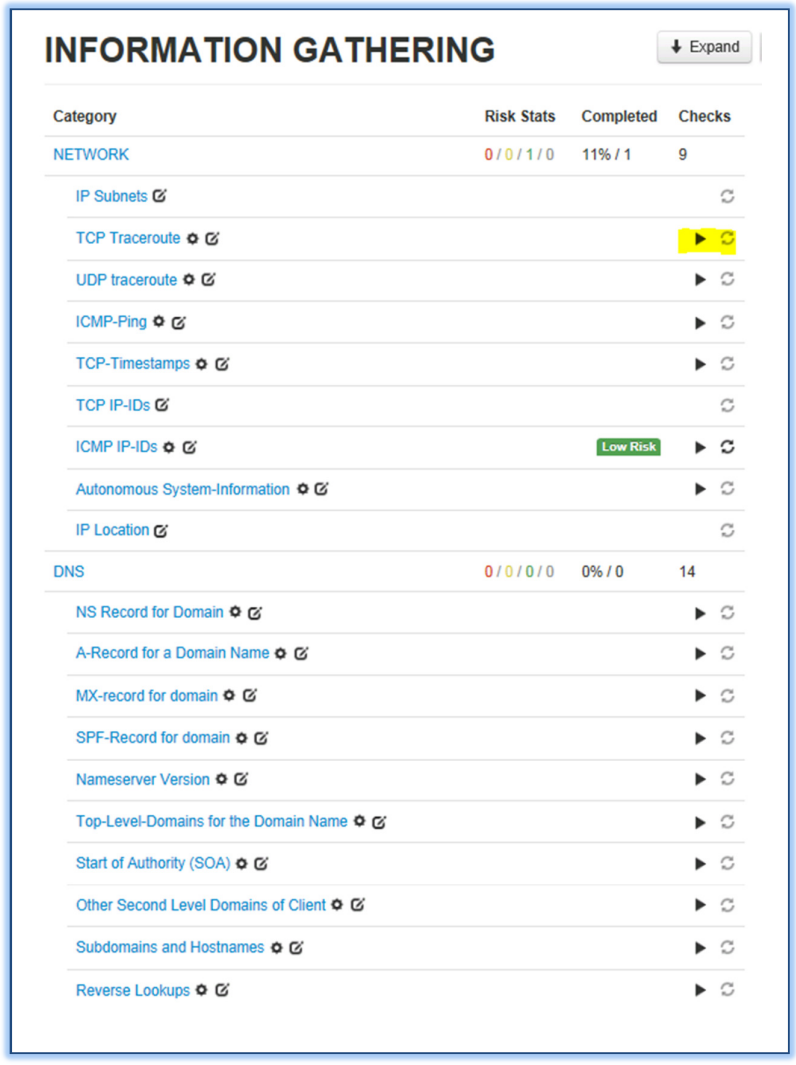
(e.g. when doing a pure information gathering).

Start a project as an auditor

In the overview page of the project you will then see all your targets and how many checks are already finished within each target (the colored numbers show the amount of vulnerabilities already discovered).



When you jump into your project and click on a check category you will see first all the controls which are defined for this specific check category. You can collapse all controls at once or just open a specific control set. After opening a specific control sets you see the actual checks you can do. The checks which are fully automated have a "play" button next to its name. You can start each single automatic check by clicking the play button or at the top menu you can click "start" and it will invoke all automated tests within all controls at once.



Once you started filling out the results and allocated a risk you will see which checks have been already finished. The "refresh" button is used if you need to re-do a specific test again.

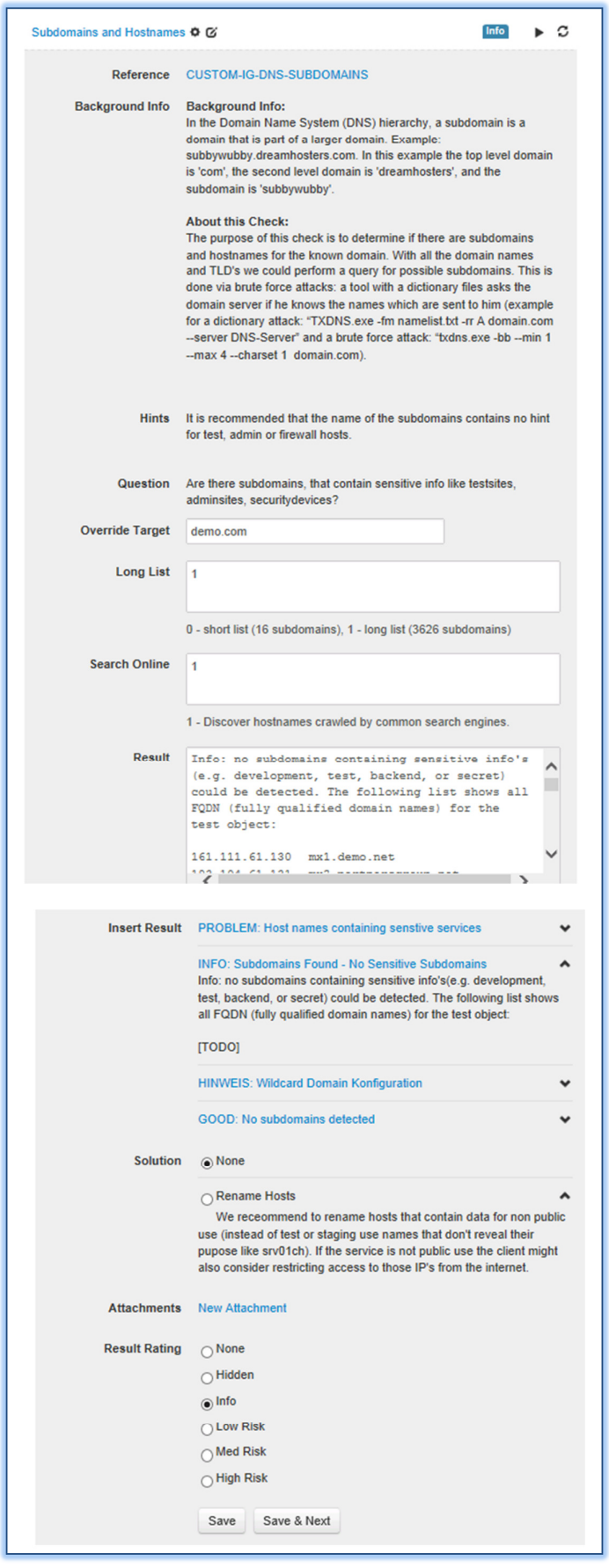
When opening a check the test user will get the fields displayed which were defined by the admin:

- **Result/Insert Result:** the result field is the actual place where the tester has to put in his results or the automated tool will display the output. Within the check the user can now also select the predefined results or solutions that apply for the discovered vulnerability.

- **Attachment**: the attachment button allows the user to add screenshots, which will be automatically added in the report later. Last not least the tester has to choose the risk level.

- **Override Target**: for automatic checks you could manually override the target (defined at the beginning of the test) to do multiple automated checks in a row. The result will always be appended to each target without deleting the prior result.

- **Options**: tool options as defined from the administrator will appear with their default settings. The auditor then can change them as he wants. In this example he can choose to run a subdomain check either against 16 or 3226 host names in the database and also include an online google analysis (google will be queried against the domain name and all unique subdomains will be filtered and inserted in the results).



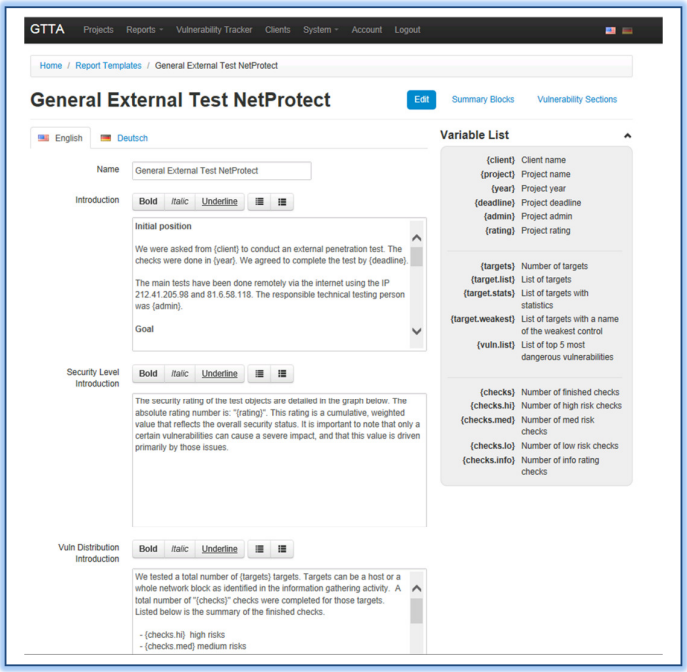
Create Reports

GTTA offers various report types. But before creating a report you will need to define a report template. The template is separated into different chapters (like introduction, report details, management summary etc.). For each chapter you can allocate your own text blocks. Additionally GTTA offers variables. Those variables pull out information's from the database and put them into the report. This allows the creation of individual reports. Some variables like {rating} will create graphics and diagrams – others report parts like degree of fulfillment, risk matrix etc. can be chosen to include or exclude in a later stage when actually creating the report.

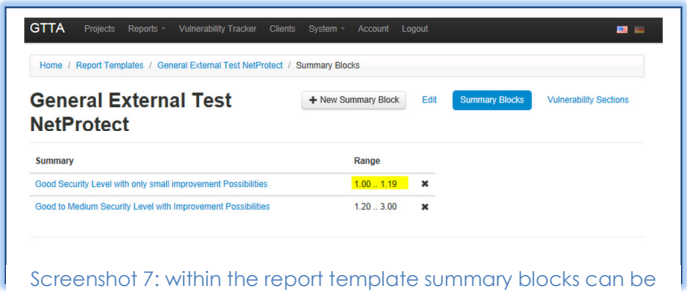
GTTA offers as well some sort of an expert system when it comes to present different management summaries based on the findings. Within the report template you also find "summary blocks". You can define an overall risk rating for a penetration test and allocate a specific text to such a rating (e.g.: if the rating is between 1-1.87 then you can tell GTTA to choose a specific management summary introduction for this type of security level).

After defining a template you can create different type of reports in GTTA (project reports, project comparison, vulnerability exports etc.).

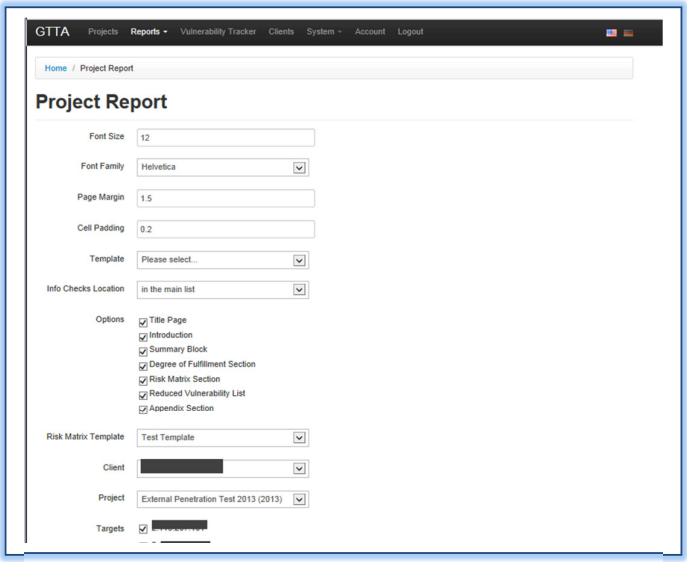
- **Project Reports:** this is the standard report which will include an overall security rating & a rating for each test object. All checks that have some results will be included in the report. The security tester can include all test reports or only dedicated targets in the report.



Screenshot 6: Sample Report template with dynamic input variables

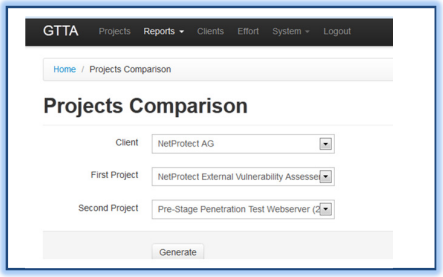


Screenshot 7: within the report template summary blocks can be dynamically allocated to specific overall ratings

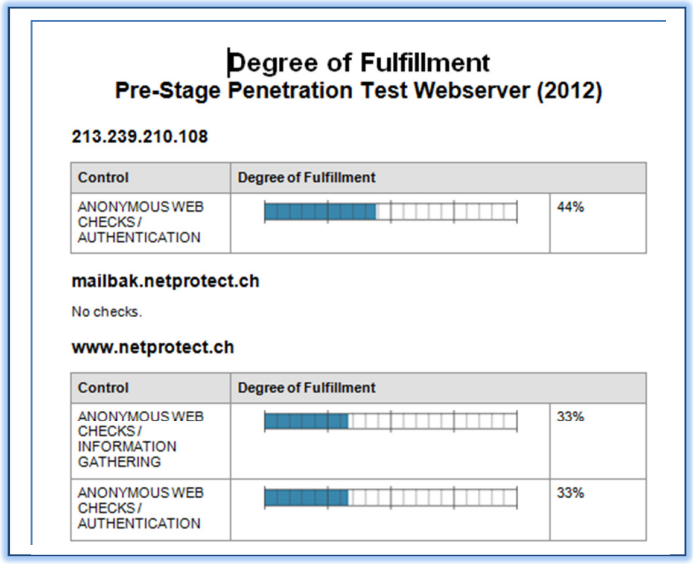


Screenshot 8: when creating a project report the user can set specific formatting options and select the targets he wants to be included within this report

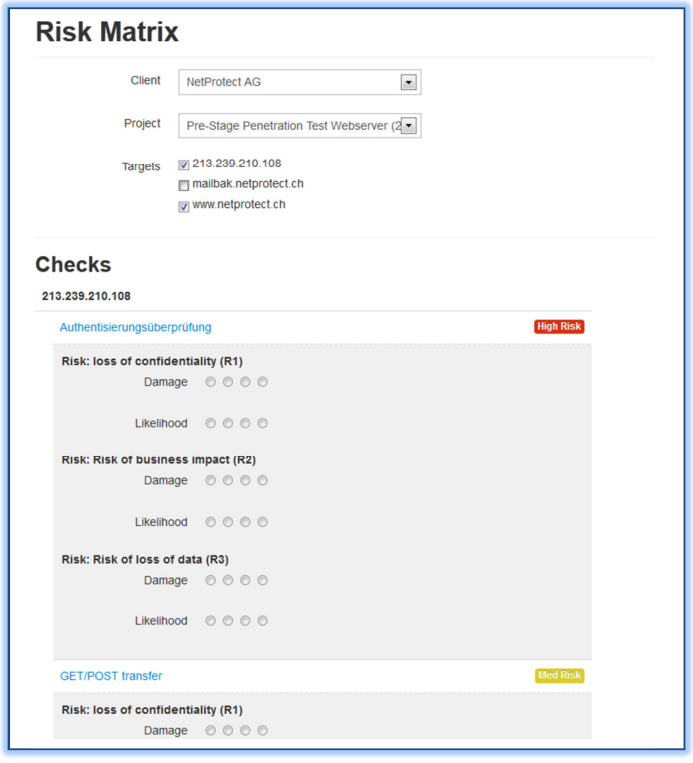
- **Project Comparison:** this type of report can be used for ongoing re-tests. It helps compare the risk ratings for the same test objects. It will automatically detect the common targets and compare the amount of vulnerabilities.



- **Degree of Fulfillment:** In the input fields of each check we defined controls. In this section, graphs are provided to demonstrate the fulfillment of the individual of security controls. Security controls are safeguards or countermeasures designed to avoid, counteract or minimize security risks. The degree of fulfillment shows the security level for the individual controls with zero being the worst value and 100 indicating all security controls are met in accordance with the checklist.



- **Risk Matrix:** the risk matrix is created out of all checks from a project with medium & high risk. The user will be presented all checks in this rating category. The user then is able to adjust manually a risk classification for each check (pre-defined risk ratings are stored, but can be adjusted). Additionally he can to allocate a likelihood AND a damage for each risk classification. The user can choose multiple risk classifications for each check (the risk categories are defined in a separate menu under system/risk templates accessible only for the



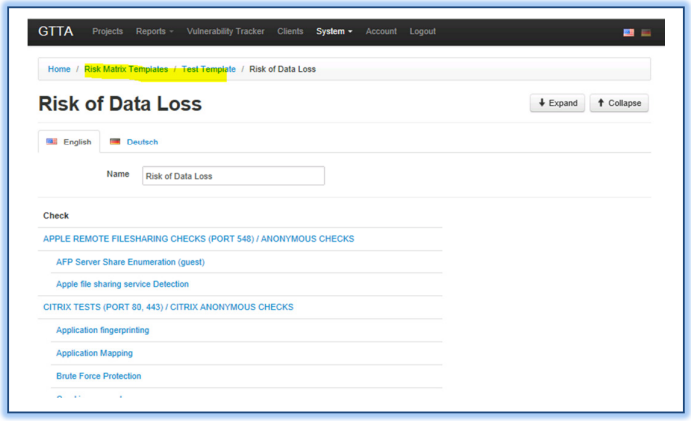
Screenshot 9: when creating a risk matrix report the default template will be applied to each finding. Before printing the report the user can still customize the risk rating

admin user).

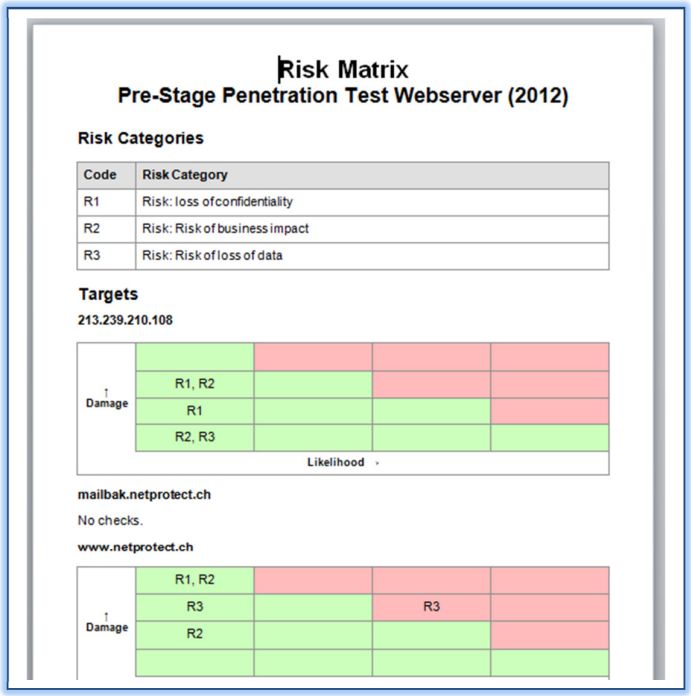
Once the risk settings are adjusted the user can create a report where the risks are displayed (see automated report output on the next page) for each target.

- **Effort calculation:** this helps the sales person or the client to calculate the estimated time. It can be done as follows:
    1. Choose test category (here the existing test categories are shown)
    2. Choose the number of targets per test category (e.g. 3 IP's within SMTP test, 2 IP's within web test etc.)
    3. Choose Test type (basic vs. advanced, only OWASP or only OSSTMM or only etc.)

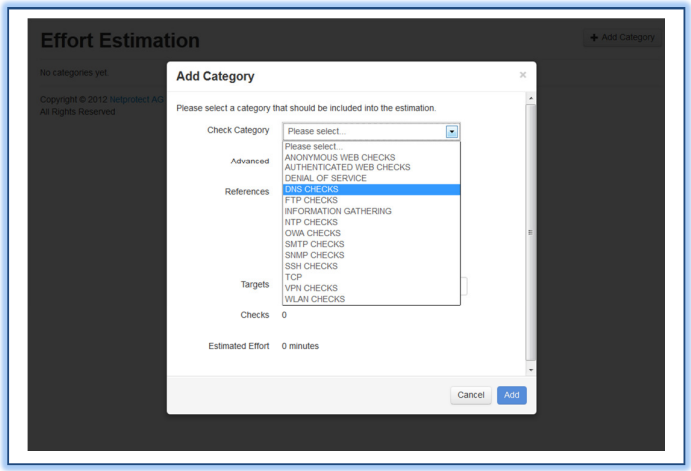
Based on this the user will be presented a table with the estimated time for the project he could print out or show to his client.



Screenshot 10: allocating a previous defined risk to the actual tests



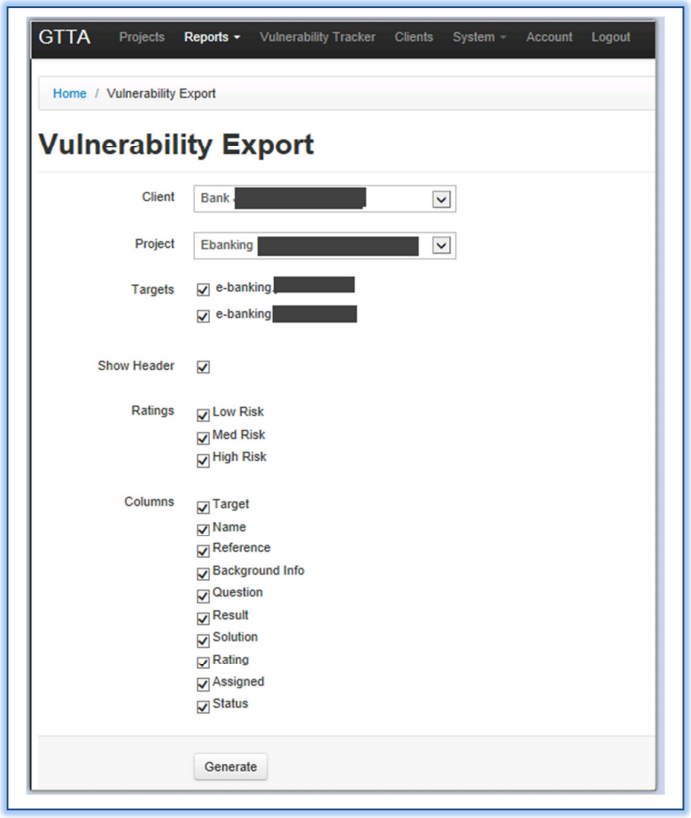
Screenshot 11: Risk matrix



Screenshot 12: Effort calculation

- **Vulnerability Export (reports/vuln-export)**

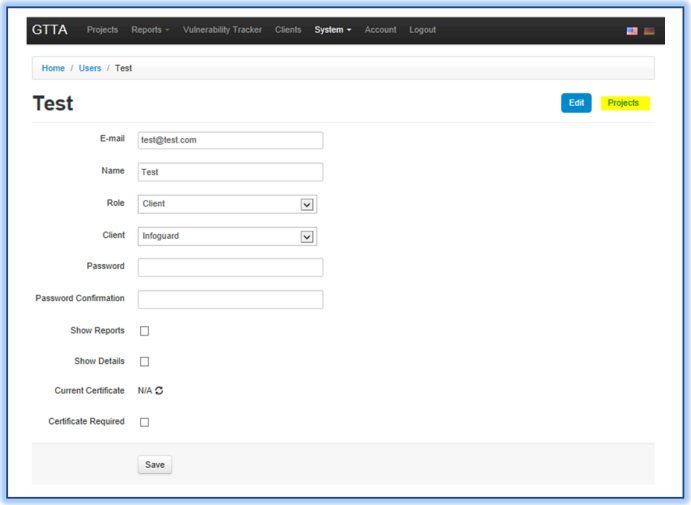
  Some clients are only interested in having a simple Excel file where only the vulnerabilities combined with a few extra info's are presented. Therefore GTTA has an export function that can export only the vulnerabilities found. The user has the ability to define, which type of results should be included and which fields. For example the user might want only an excel with all high & medium risks found, only with the check name, target, results, Recommendation & Rating.



Users (system/users)

We have three user types:

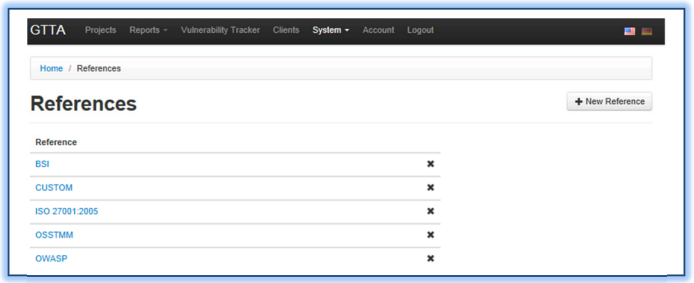
- **Admin**: can create checks, references etc. and can conduct tests.
- **User**: the actual testing person – cannot edit checks
- **Client**: project managers or clients. They will only see the summary of the test results and its progress (how much is already tested, how many vulnerabilities discovered etc.)



Additional options can be set for the users:

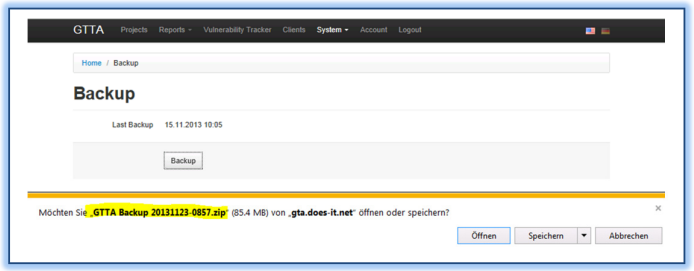
- **Authentication:** password and/or certificate based authentication (GTTA creates a custom certificate for you with one click).
- **View/edit rights:** if you create a new user as a client or standard auditor you can restrict him by only allocating specific clients and projects to him.
- **Notifications:** Admins and standard can get automated notifications (e.g. if GTTA has finished an automated test)
- **View option for clients:** you can additionally restrict clients view options by allowing him to see only the vulnerability names, full vulnerability details, full report etc.

References (system/references): References (like ISO, BSI, OWASP) are defined with a specific URL in this section. You can create your own custom references which will be needed to be associated with each specific check. References can be used when staring a new project: you could e.g. select the reference "OWASP" for a specific target and GTTA will only present the auditor with all checks marked with the reference OWASP.



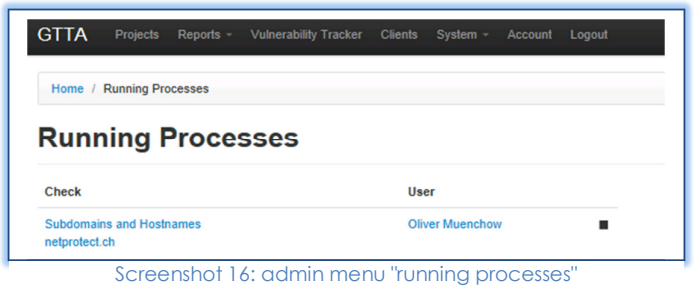
Screenshot 14: admin menu "references"

Backups (system/backups): GTTA will automatically back up all data in the background. You can additionally manually backup your data (including client data and all checks or templates you created. All data can also be restored easily (system/restore).

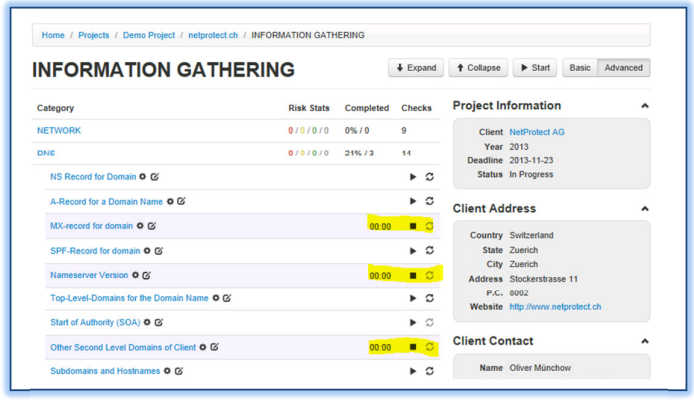


Screenshot 15: admin menu "backups"

Running processes (system/processes): GTTA gives the admin the possibility to view and if needed to stop all running automated tests. The same feature is available within the actual project from an auditors view: every auditor can see which tasks and how long they are currently cunning for a specific target. The auditor will be able to stop them as well in case he needs to.

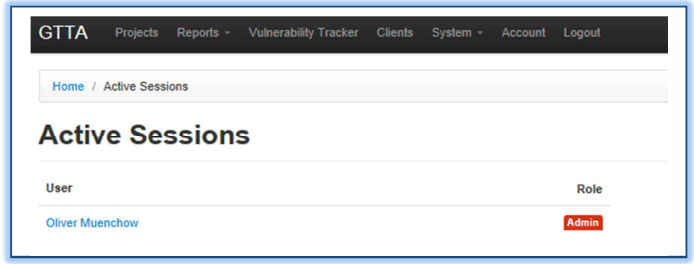


Screenshot 16: admin menu "running processes"

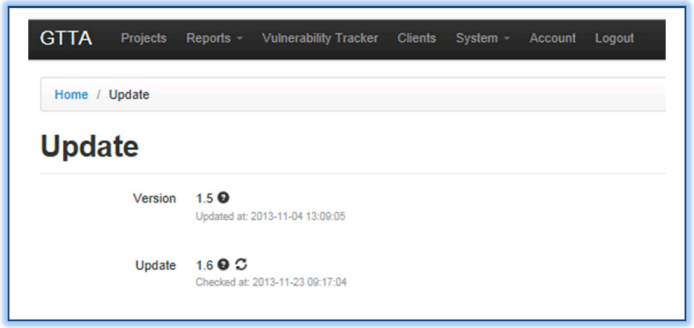


Screenshot 17: processes can also be viewed/stopped from within the project

Active sessions(system/sessions): GTTA offers the admin the possibility to view who is currently logged in.



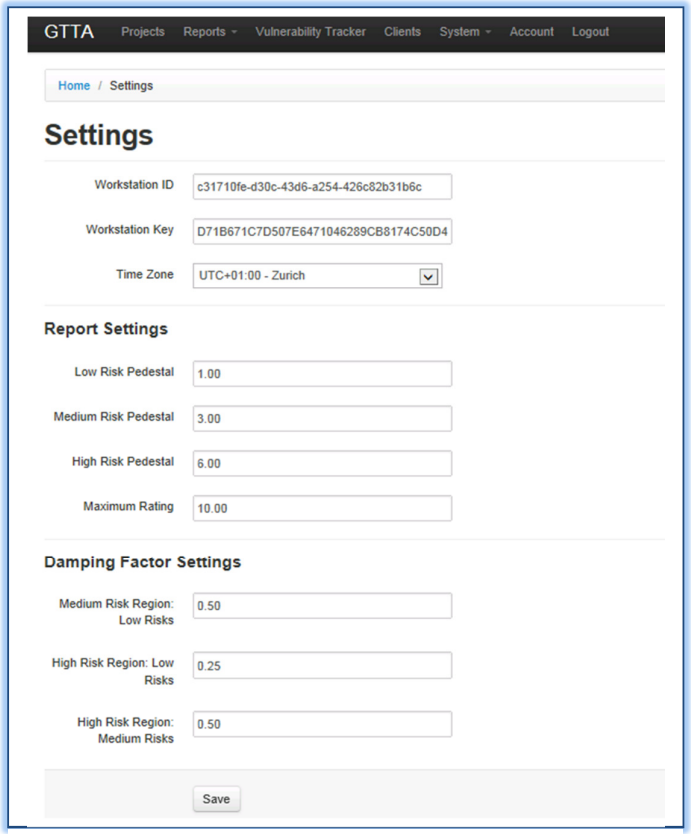
Updates (system/update): GTTA checks automatically for new updates. Updates can include new checks but also new features or system changes. The update process is fully automated and can be executed with one click.



Screenshot 18: Update settings

Settings (system/settings): beside the time settings and workstation license you can alter the report rating formula here (GTTA calculates risk ratings from 1-10 for the whole project or for a specific test object based on a formula).

- First, a **pedestal** number is established. This is the minimum value the overall rating can have given the number of vulnerabilities found. It is set to 1 if there are only low risk vulnerabilities, to 3 if there are medium risk (but no high risk) vulnerabilities and to 6 if there are high risk vulnerabilities.
- Next, a **dynamic maximum** is calculated, which gives the maximum value given the number of vulnerabilities fund. It is 10 if there are high risk vulnerabilities, 6 if there are no high risk but medium risk vulnerabilities and 3 if there are only low risk vulnerabilities.



Screenshot 19: general system settings and report formula

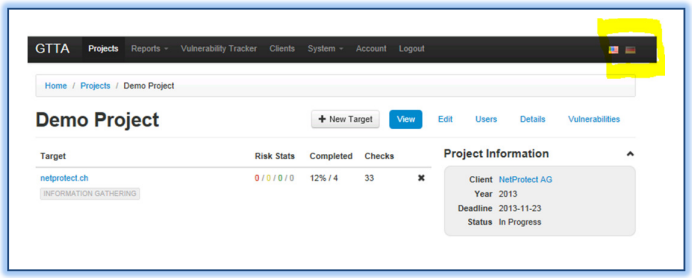
Within these limits, the rating is determined in the following way: the fraction of highest risk vulnerabilities found (i.e. vulnerabilities found divided by tests performed) is multiplied by the remaining "space" (i.e. dynamic maximum - pedestal), giving the contribution of this risk class. Then, the remaining "space" (i.e. dynamic maximum - [pedestal + contributions of previous risk class]) is multiplied by the fraction of vulnerabilities found in the next lower risk class, multiplied by a damping factor (explained below). This step is then repeated for the lowest risk class.

The **damping factors** reduce the contribution of lower risk classes in case a higher class vulnerability is present. This means, if there is a high risk vulnerability, the medium risk vulnerabilities can fill up a maximum of 50% of the remaining "space", and the low risk vulnerabilities thereafter can only fill 25% of the remaining "space". In case there are no high risks, but medium and low risk vulnerabilities, the low risk vulnerabilities will only fill

50% of the remaining "space". The behavior can probably be best explained by an example:

Example: consider the case where 10% of the medium risk tests an d 100% of the low risk tests have uncovered vulnerabilities. If there were no damping (i.e. damping factor for low risks in medium region = 1), the pedestal would be 3, the contribution of the medium risks would be 0.3, giving a total so far of 3.3 and the low risks, where 100% failed, would fill up the remaining "space", giving a final overall rating of 6, which would probably be considered too big. With a the corresponding damping factor of 0.5, the contribution of the low risks in the medium range will be halved, giving an overall rating of 4.65, which seems more reasonable.

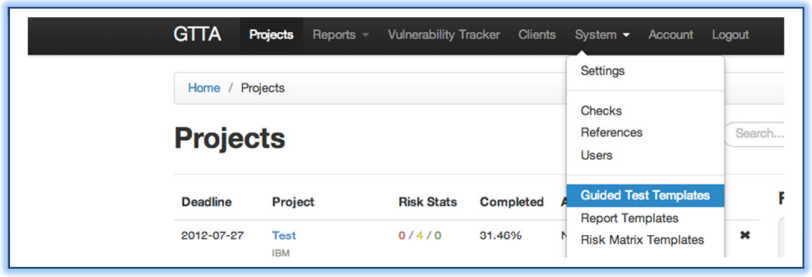
Languages: Currently all checks & menus are defined in English & German. By clicking the language flag you can quickly change the current language. Other languages can be added as well. Selecting the language in the navigation menu also affects the final report language. Therefore it should always be selected first. Other languages can be added on request.



Creating Guided Test Checks

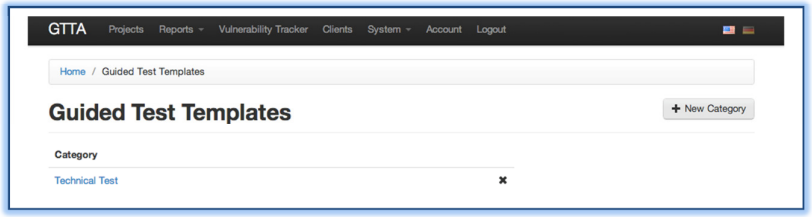
First of all, you will need to create some Guided Test checks in order to be able to use Guided Tests for your project.

1. Click on "System -> Guided Test Templates" menu item



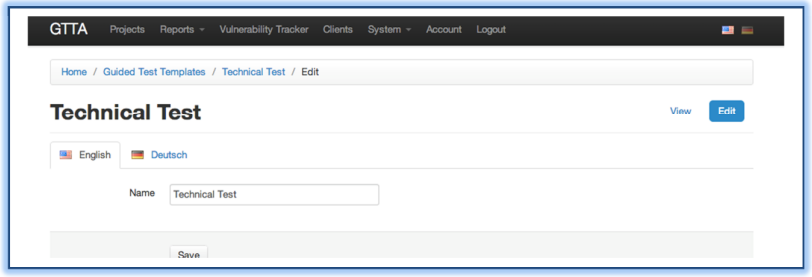
2. You see a list of Guided Test categories on this page. Press "New Category" button to create a new category.

3. Enter the desired category name and hit "Save" button, then click on the "View" link for the created category (this link is in the top right corner of the page).

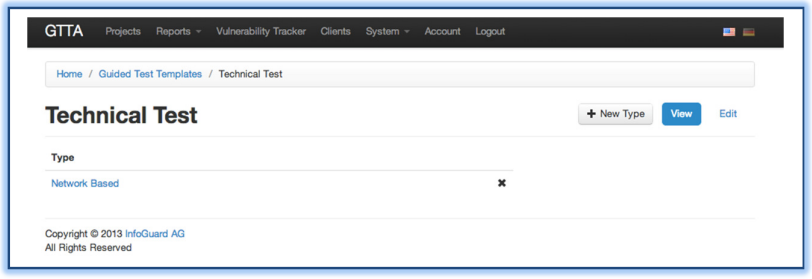


4. Now you see a list of types for the category. You can create a new type by pressing the "New Type" button.

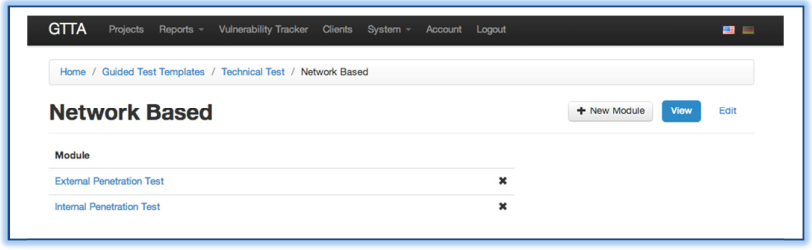
5. On this page you should enter the desired type name and hit "Save". After type is saved, please press the "View" link in the top right corner of the page.



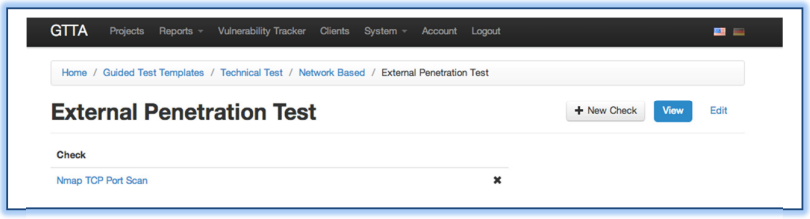
6. Now you are viewing a list of modules within the type. You can create a new module by hitting the "New Module" button.



7. After creating a new module, please click the "View" link.

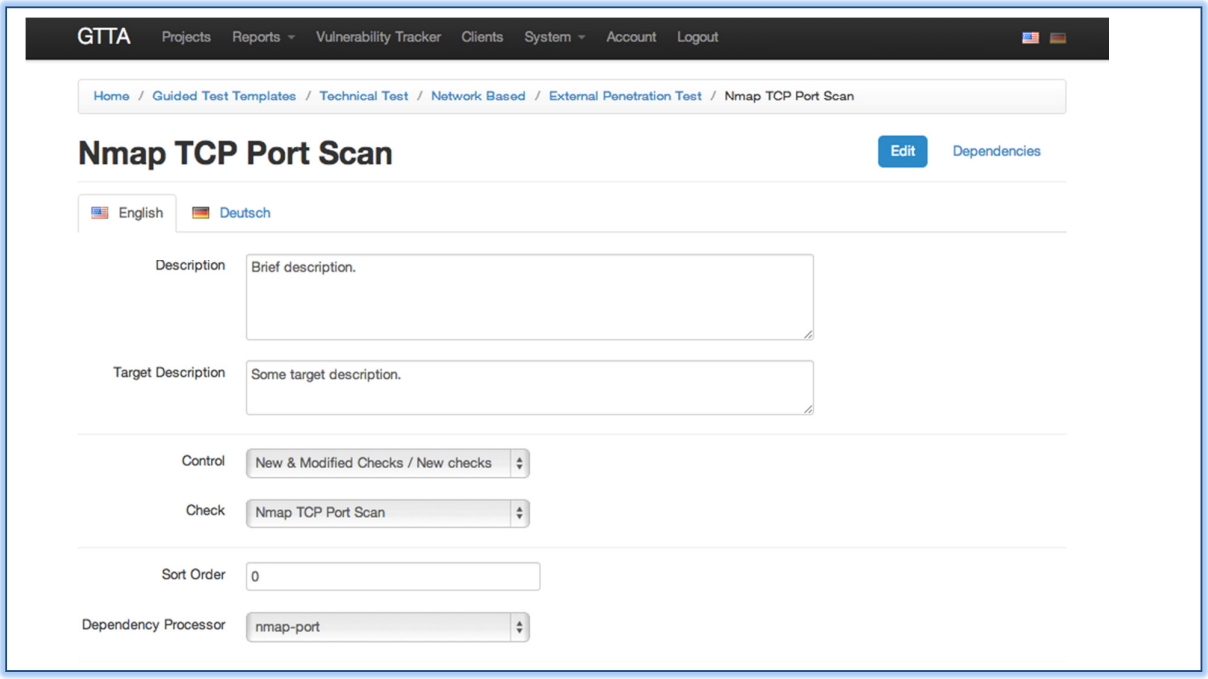


8. Here you see a list of checks within this module. You can add a check by pressing "New Check" button.



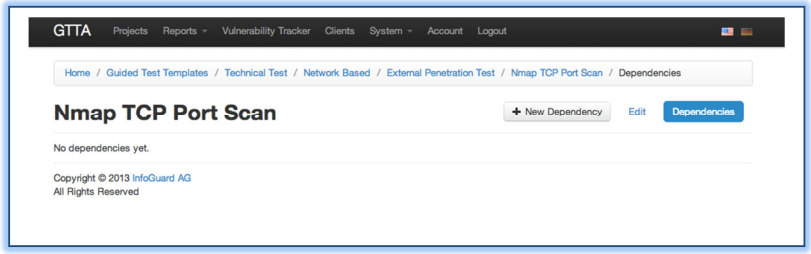
9. On the check creation page you can enter the following information:

   a. **Description** – current check description
   b. **Target Description** – a brief target description for this check
   c. **Control** – a check control that contains the desired check
   d. **Check** – check that will be added for this module
   e. **Sort Order** – check sorting order within a single module
   f. **Dependency Processor** – software that will handle check dependencies for this check. There is only one Dependency Processor available at this time – "nmap-port" – it reads nmap output and suggests targets for other modules. Choosing this Dependency Processor only makes sense when you select "Nmap TCP Port Scan" check, for other cases use "N/A".



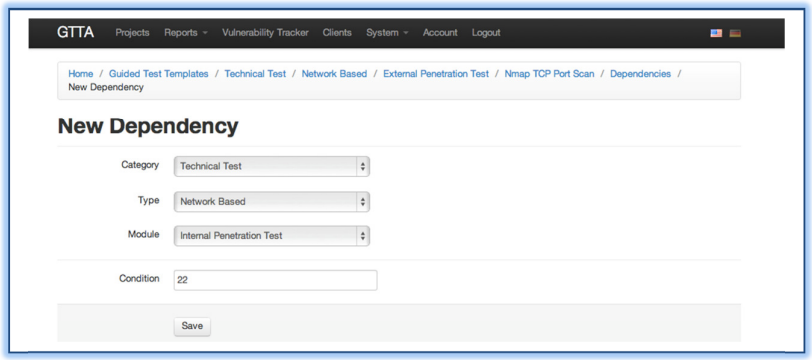
10. If you selected some Dependency Processor for the current check, you will need to define some dependencies for it. Click on the "Dependencies" link on the top right corner.

11. Now you are on the page with a list of



dependencies for the current check. Click "New Dependency" to create a new dependency.

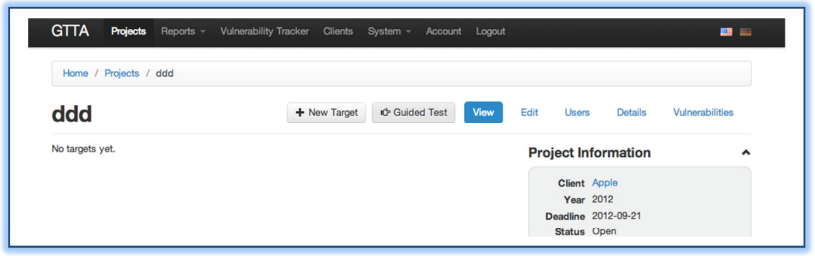
12. Here you see the dependency form:



a. **Category** – Guided Test category of the dependent module
b. **Type** – type of the dependent module
c. **Module** – the dependent module (dependency processor will suggest targets for this module)
d. **Condition** – if this condition will be true, the dependency processor will suggest new targets for the dependent module. For "nmap-port" dependency processor this field should contain a port number that should be open. If this port on the target is open, then the dependency processor will add a new module for the project (if it is not added yet) and suggest the target for it.

13. After saving the dependency, you can go back to the dependency list. You can create as many dependencies as you wish. Now you're all set and can use guided tests.

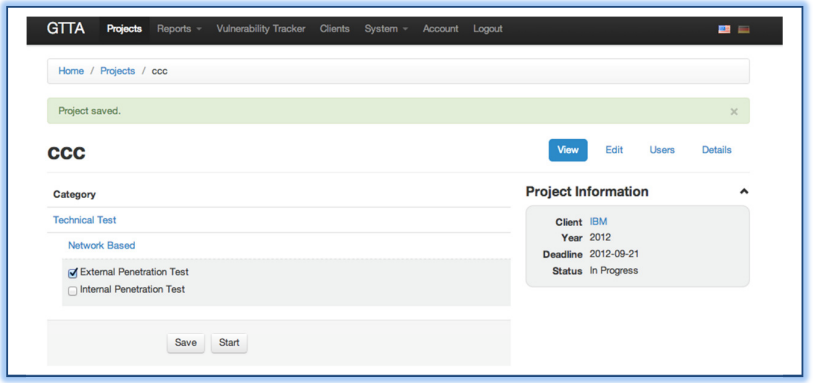
Performing Guided Tests



You can run guided tests on any project if it has no check results or attached targets yet. Projects with guided tests are displayed with "hand" icon in the list of projects. Here is a list of steps for running guided tests in a project:

1. Open an empty project or create a new one. You will see a "Guided Test" button on the top – click it to turn the project into a Guided Test project.

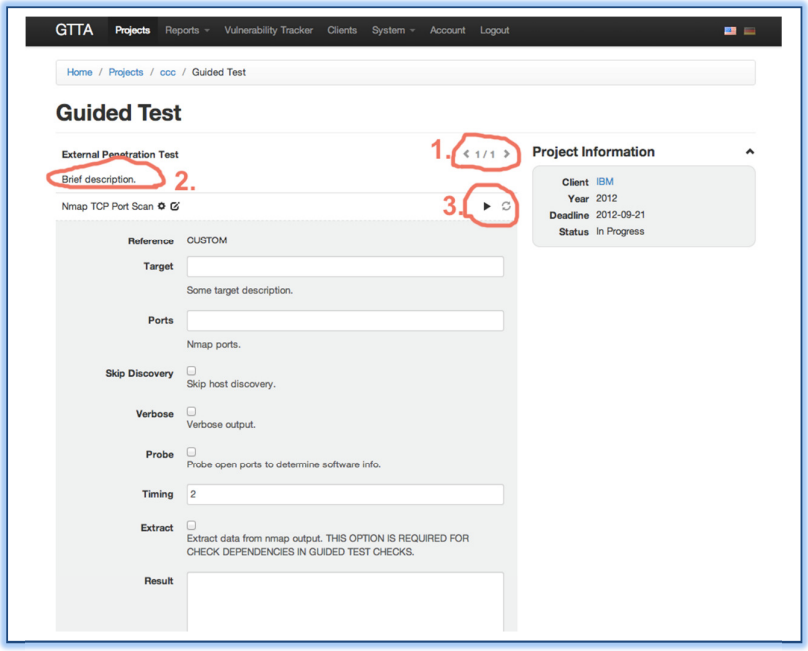
2. You will see a module selector for the Guided Test. You should unfold the desired categories and types and select modules that you wish to run for this project.



3. After you select modules, please press the "Save" button below. If you selected

any modules, you will see that the "Start" button will appear next to the "Save" button. You should press it to start Guided Tests.

4. Now you are on the Guided Test Check page. It's very similar to usual check page, except some differences. The main difference is that Guided Tests display only 1 check per page. Also there are some additional controls:

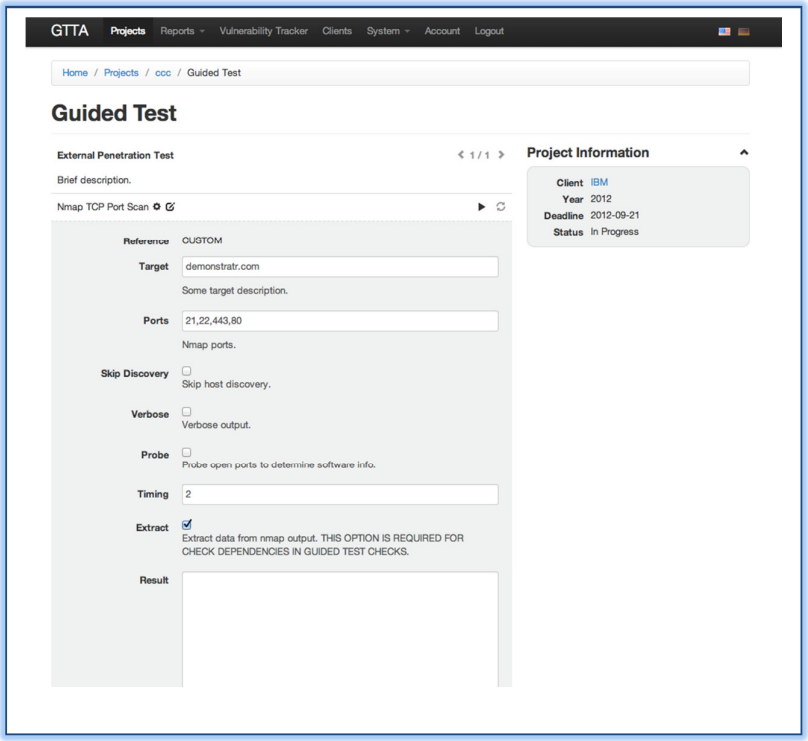


- **Check controller**. You can press "Back" or "Forward" buttons to navigate through checks within all selected modules. Numbers here display the number of the current check and the total number of checks.

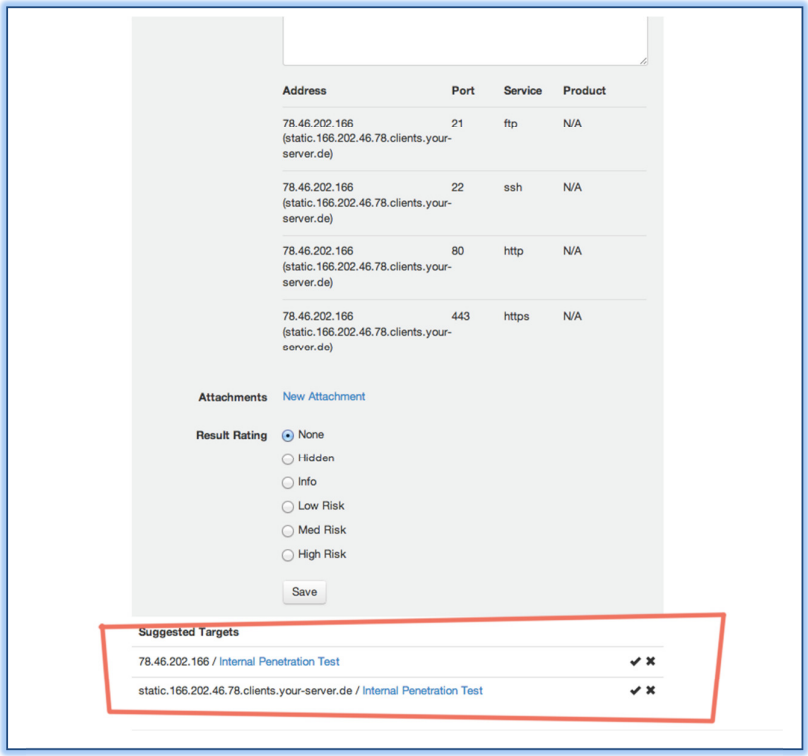
- **Brief task description**. You can add or change this text on the module's check edit page (under "Guided Test Templates" menu).

- **Standard check controls**. You can run or clear the check contents using them.

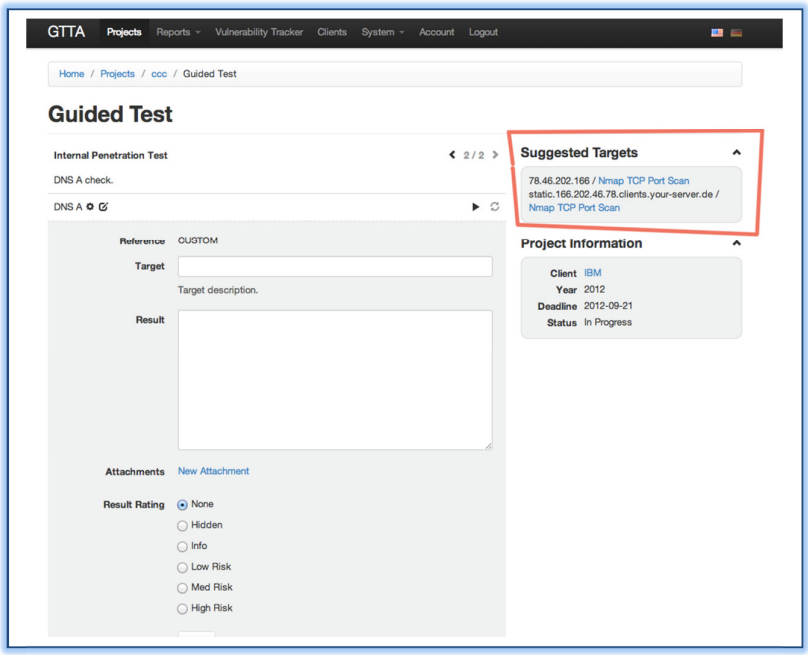
5. Enter some test details here, as if you are on the usual check. In the example on the screenshots I'm filling in the Nmap TCP check, so I will be able to demonstrate you how check dependencies work. Please note, that if you want to use check/module dependencies you should set the "Extract" option of the "Nmap Port Scan" check. Otherwise check dependencies won't work.



6. After running the Nmap check, the check suggests several additional modules and targets with open 22nd port. If you accept the suggestion, you should click on the "✔" icon, otherwise press the "✖" icon to delete the suggestion. After dealing with all suggested targets (if any), you should refresh the page and see if the system added any additional check modules to the current project. In our case, the system adds "Internal Penetration Test" module to the project with 1 check in it, so you can press "Next" button in the Guided Test Navigation menu.



7. Here you will see that the system suggest you 2 additional targets. It also specifies the check that suggested these targets. If you wish, you can copy & paste these targets into the target field and perform all required checks, otherwise you can go to the check that suggested these targets and delete the suggestions.



Metasploit

There is only 1 script responsible for the whole metasploit integration. In order to create a metasploit check script, please do the following:

1. Create an automated check (or choose an existing one) that will hold the metasploit script.
2. Go to the *Scripts* section of that check.
3. Press *New Script* button.
4. Choose **metasploit** package in the package list and hit *Save*.
5. Now you need to create an input that will hold the metasploit commands. Please go to the *Inputs* section of the script you just created.
6. Hit *New Input* button.
7. Setup the input that will hold the metasploit script:
    1. *Name* - the name doesn't actually matter - you can enter anything there (for instance, it can be named as **Script**).
    2. *Type* - set it to **Textarea**
    3. *Value* - enter metasploit commands here - the same commands that you would use for msfconsole command in metasploit. Please note, that the system automatically adds **run** and **exit** commands at the end of each script, so the script will be launched automatically.

       You can use some variables in your script:

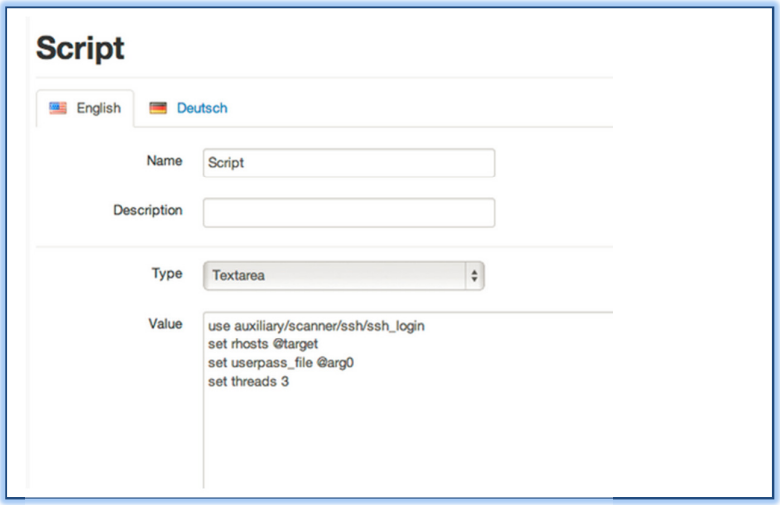
       - **@target** - check target
       - **@argN** - a file with values from the other input. **N** here is a number starting from 0, so **@arg0** would be the first input, **@arg1** would be the second input and so on.

       Here is the example script that does a SSH bruteforce:

```
use auxiliary/scanner/ssh/ssh_login
set rhosts @target
set userpass_file @arg0
set threads 3
```

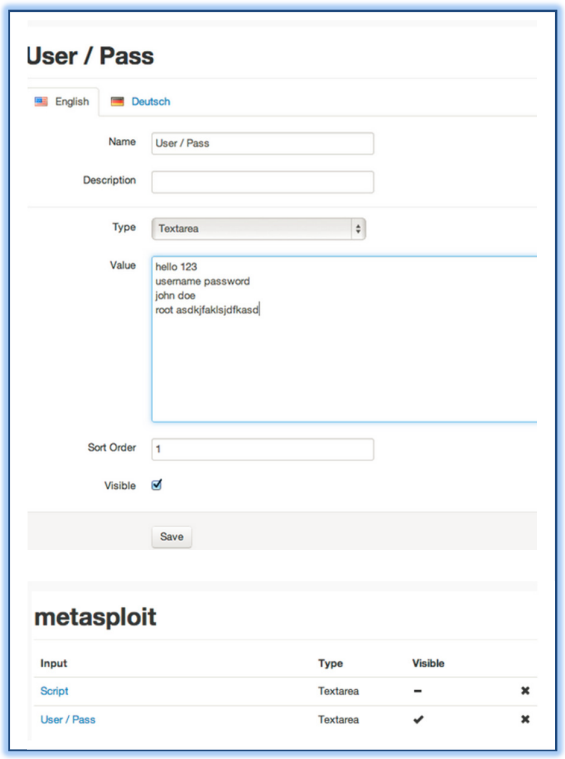
4. Visible - unset the checkbox, so the in put won't be visible in the checklist.

   Then hit *Save*.



8. If you have used some additional arguments in your script (**@arg0**, **@arg1**, etc.), then you will need to add corresponding inputs to the script. For our example script above you need to specify an additional input that will hold login/password pairs. Please refer to metasploit documentation to find out the required file formats for each module.

9. Now it's time to go to your project, use the check you just created and try to start the script.



Hint: if you need to set a script with a path: here is an example of a build in namelist: **/opt/metasploit/apps/pro/msf3/**data/wordlists/namelist.txt