EMREX

Security Testing Plan

Introduction

A penetration test for the EMREX will give the stake holders information on the current security level of the application, and it will be easier to make decisions on which measurements are necessary to get the right level of security.

Test environment

To get the most out of the test as possible it's important that the test environment is as similar to the production environment as possible.

The purpose of a pen-test

The purpose of a pen-test is to check applications for security vulnerabilities. where destroying data or causing servers to crash does no
real harm. The result of such a pen-test is a report which for each vulnerability contains:

- Type of security vulnerability
- Explanation of this kind of security vulnerability and how such vulnerabilities can be used to exploit your systems
- Categorization of security vulnerabilities in the application being tested
    - Critical: it is our opinion that the application can be compromised and the application should not go to production.
      We notify the system owner immediately.
    - High: we highly recommend that these kind of vulnerabilities are fixed before production.
      We notify the system owner after the test is finished
    - Medium: we recommend that these kind of vulnerabilities are fixed before production.
      We notify the system owner after the test is finished
    - Low: a security vulnerability that can be dealth with when you have the time.
      We notify the system owner after the test is finished
    - Info: not nessesary a security vulnerability but needs improvement

The categorization of security vulnerabilities are only advisory and should be used as such in a risk assessment of the system.

Preparation before a pen-test

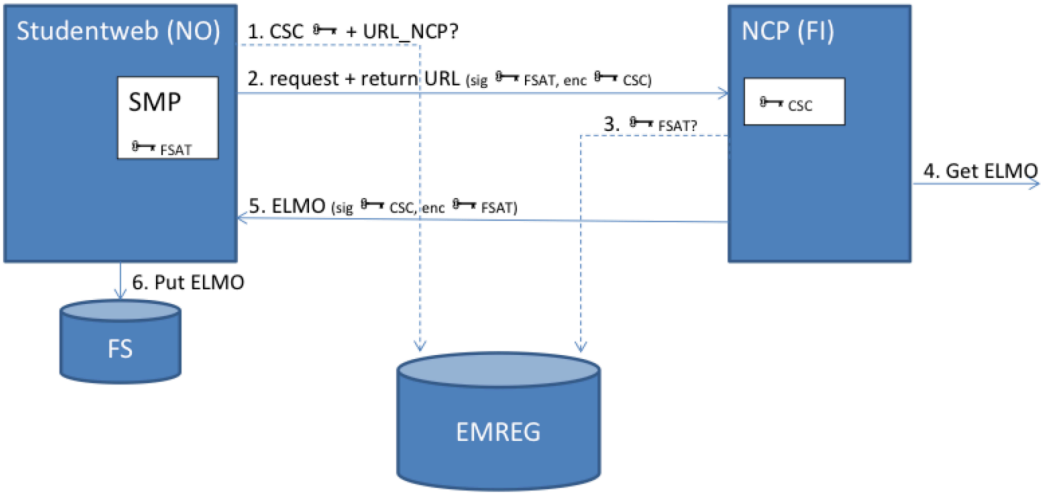
Several things must be done before the test can be conducted:

- Access to a user of each access level used for working in the application and to try privilege escalation
- A test server which has the application up and running working as it would in production. This way it's easier to test all of the functionality of the application.

- We need access to the application, e.g usernames and passwords, ip's to access if ip control
- It's smart that we get a person to contact if we get in trouble with database access, access to the application or something else
- It's smart to turn off firewalls or waf's or similar tools. We want to test the application, not the firewall and waf. Firewalls and waf's should be seen as security in depth mechanisms.

Scope of test

When performing a test against EMREX we test the data flow shown in the figure.



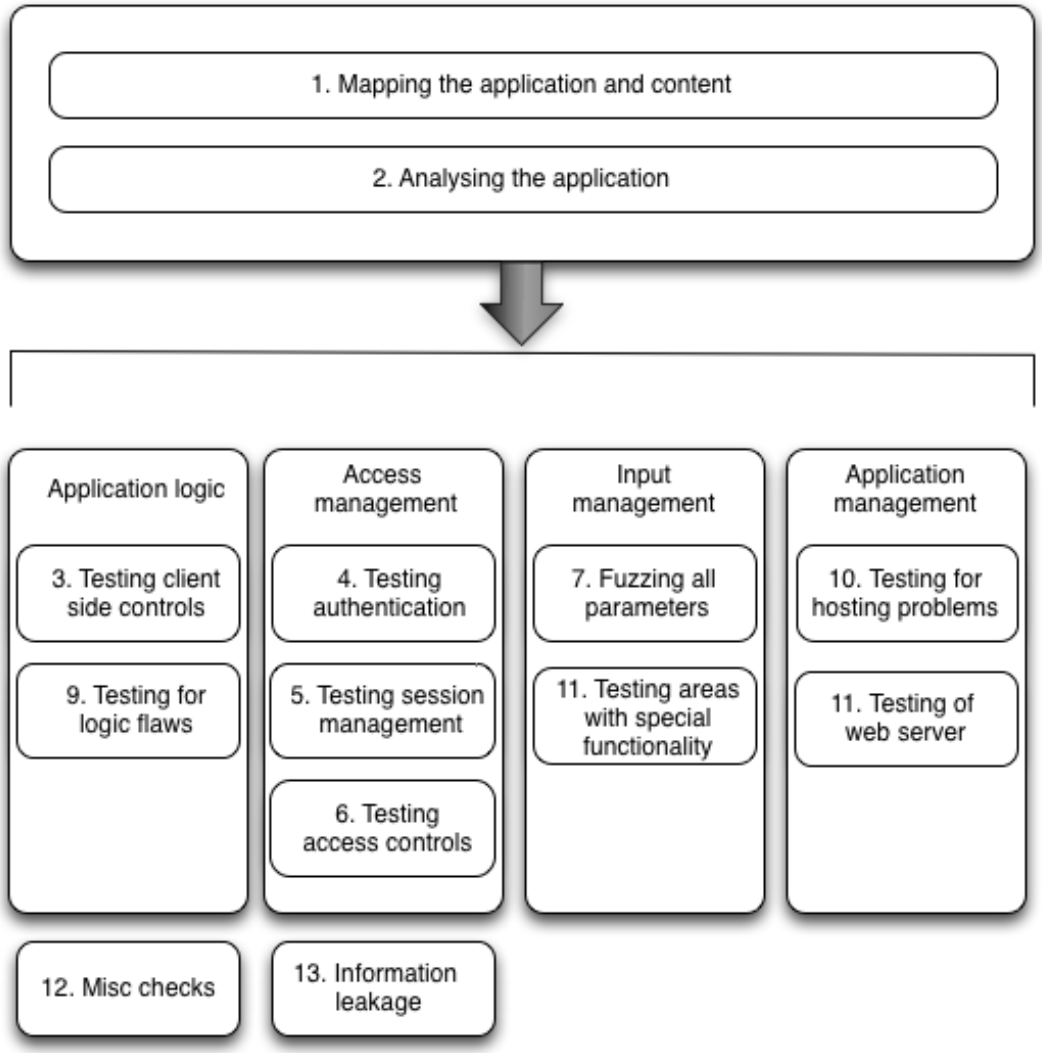
Id	Type	Name	URL_NCP	Pubkey
1	SMP	FSAT		ssh-rsa AAAAB3NzaC1yc...
2	NCP	FSAT	http://fsat.no/	ssh-rsa AAAAB3Xgfe3vde...
3	NCP	CSC	http://virtawstesti...	ssh-rsa AAAAmder3tgfew...

The plan is to test the security in all 6 movements of data in the picture above:

- Is it possible to manipulate the list of NCPs?
- Is it possible to manipulate a request to a NCP?
- Is it possible to "man in the middle" somewhere?
- Is it possible to manipulate administration of the EMREG site?
- Does the solution have end-to-end encryption?
- Other tests

Tools and methodology

We will use Open WebApplication Security Project (OWASP) principles to search for vulnerabilities. We will use open source tools like Burp proxy, sqlmap, nmap, ncat, nikto and other tools provided by Kali linux. The figure below shows the work flow of a pen-test.



Estimated use of time

A good pen-test of an ordinary web application takes approximately 2 man-weeks, but it's possible to eliminate low hanging fruit and some more in 1 man-week for each web application.

We have two web application pen-testers available and need to know about a pen-test at least two weeks in advance.

Database

It's highly possible that we destroy the data in the database and it's highly recommendable that it's possible to restore the database when needed.

Whitebox testing

Having application source code on hand makes vulnerabilities easier to find. Full code review is also possible. It can reveal more vulnerabilities but is very time-consuming.

Black box testing

A black box test is conducted without access to source code. The pentester does not know how the inside of the application works and has to observe how it reacts to different input. This is more realistic as it is similar to how an actual attacker would operate. But it is less useful when the goal is to uncover vulnerabilities.

When to run a pen-test

All applications should be tested for security issues. Penetration testing should be performed on a periodic basis depending on the importance of the targeted system. We recommend running a pen-test on an annual basis or after any major systems upgrades or changes.