FTA – Web Application Security

Luboš Bretschneider <bretik@gmail.com>

Abstract

This document describes process of Fault Tree Analysis (FTA) of Web Application Security. Suggested fault tree graphically represents main security risks and probability of misusing security holes in the application. It describes most common security issues and causes of system failure. Failure in this context means security breakthrough, or unauthorized access to any protected part of application.

Process of creating Fault Tree

To create fault tree, I divided the identification process to four parts. In first part, I will identify the Top Level Event, it means event, which is critical for our application. I will analyze this event and try find out the causes of this event.

In second part of process, the aim is to find the faults, which can lead to the Top Level Event – it means the main, root mistakes, which can directly threaten our Web Application.

Third part of analysis is part, where I will try to identify the causes of root failures and the relations among them. There are two possible relations: relation "*and*" among causes means, that all of them has to happen to cause failure. Relation "*or*" among causes means, that at least one of them has to happen to cause failure.

In the last, fourth, part of analysis, I will use all previous parts to construct fault tree, which is graphical representation of previous analysis.

Top Level Event

As a top level event, we'll consider Web Application Security Breakthrough. Security Breakthrough means, that intruder gained access to server, or application parts, which he is not allowed to access. For Web Application this means, that attacker can make some unauthorized changes to administration of application, or change some server settings – this can lead to server, or application crash, or to misusing application to advantage of attacker.

Faults that could lead to the Top Level Event

All the main and important faults that can lead to system failure (security breakthrough) are listed below with sort description. These faults were identified by analysis of the Top Level Event.

Unauthorized access

Access to non-public parts of application, could be really dangerous. If some attacker is able to gain unauthorized access to administration of application, he can use it to his advantage and make some serious damage, starting with administrating user accounts and continuing with altering our database and deleting some important data.

External Content Execution

Execution of any external content on our server means big risk. By executing external content, attacker is able to make serious damage to our server and he can gain total control over our application, its files and database.

Unsecured User Input Data

Unsecured user input data in our case means any data posted on server from user, which wasn't checked and altered to be secure for displaying on output of our application, or for using in application code.

Causes of faults

Every one of the main faults named above has its causes. Next task in this analysis is to identify these causes and relations among them.

Unauthorized access

There are three causes of unauthorized access – if anyone of them happen, it means unauthorized access, therefore the relation among them is “or”.

Unsecure Password

Any easy-guessable password of user is potential risk. If administrator password is easy guessable, attacker can gain access to administration part of application and use it for his advantage. When users has easy-guessable password, the attacker can misuse their accounts.

Unauthorized Access to Non-public Page

Every page, which is not checked for authorized access and is for example only hidden from the men, is not secure. In the moment, attacker can somehow get the name of the page, he can just write it in address bar and access the page without any rights.

Unsecured Cookies

Plain text passwords, usernames, or any other sensitive data in cookies means risk of unauthorized access. Any attacker, who can read users cookies, could read our data too and use it to unauthorized access to our application.

External Content Execution

There are two causes of fault called *external content execution*, both of them has to happen in the same time (relation “and”) to cause the execution of external content on our server a make some damage.

Executable Content Upload

Allowing users to upload executable content to our server is opened gate for attackers to execute it and take advantage of this mistake. In combination with execution of uploaded content below, it’s real risk for our server and application.

Execution of Uploaded Content

If some uploaded content is executed on our server, it means powerful attack. Attacker can take control over our files and database and can commit some serious damage to our application. Allowing execution of uploaded content by default is one of the most serious risks.

Unsecured User Input Data

There are four different causes of unsecure manipulation with user input data. Any of these is dangerous, but some of them are more dangerous than others. Relation among these causes is “or”, so it’s enough, when only one of them happens to cause some damage.

SQL Injection

Direct using user-readable and user-editable parameters in SQL queries, or direct using of parameters submitted by user in login form is real security hole. Attacker is able to execute addition commands on our database, including destructive queries.

Direct Page Inclusion

Direct inclusion of page written in address bar means fast and easy way, how to gain total control over files on our server. He can include his own script on his server and take advantage of it running on our server.

Input Execution

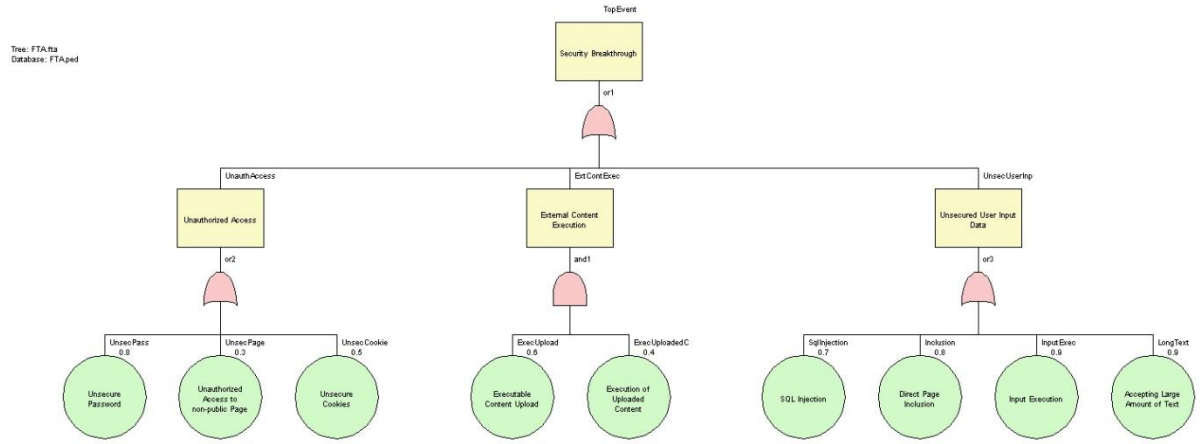
Execution of unsecured user input – it means directly displaying it to the application output – is dangerous in way of misusing our application to spam, or advertisement. Execution of JavaScript written by unknown user can cause unwanted popup windows, this can make users of our application uncomfortable.

Accepting Large Amount of Text

Accepting large amount of text as input from user can lead to heavy server load, or server crash.

Fault Tree

Fault tree is graphical representation of previous analysis. It graphically shows the relations among causes of failure and also shows the possible paths to the Top Level Event.



Conclusion

The aim of my analysis was to graphically represent main causes of system failure and relations among the causes. We can look at the fault tree as a conclusion of whole analysis – it shows all analyzed faults and causes and all important relations. Thanks to the fault tree and its clearness, we can fast and easily identify main risks and present it to other people.