# Chapter 3. How Much Security Do You Really Need?

**Table of Contents**



When one talks about security of web applications, a prudent question to pose is "how much security does this project require?" Software is generally created with functionality first in mind and with security as a distant second or third. This is an unfortunate reality in many development shops. Designing a web application is an exercise in designing a system that meets a business need and not an exercise in building a system that is just secure for the sake of it. However, the application design and development stage is the ideal time to determine security needs and build assurance into the application. Prevention is better than cure, after all!

It is interesting to observe that most security products available today are mainly technical solutions that target a specific type of issue or problems or protocol weaknesses. They are products retrofitting security onto existing infrastructure, including tools like application layer firewalls and host/network based Intrusion Detection Systems (IDS's). Imagine a world without firewalls (nearly drifted into a John Lennon song there); if there were no need to retrofit security, then significant cost savings and security benefits would prevail right out of the box. Of course there are no silver bullets, and having multiple layers of security (otherwise known as "defense in depth") often makes sense.

So how do you figure out how much security is appropriate and needed? Well, before we discuss that it is worth reiterating a few important points.

- Zero risk is not practical
- There are several ways to mitigate risk
- Don't spend a million bucks to protect a dime

People argue that the only secure host is one that's unplugged. Even if that were true, an unplugged host is of no functional use and so hardly a practical solution to the security problem. Zero risk is neither achievable nor practical. The goal should always be to determine what the appropriate level of security is for the application to function as planned in its environment. That process normally involves accepting risk.

The second point is that there are many ways to mitigate risk. While this document focuses predominantly on technical countermeasures like selecting appropriate key lengths in cryptography or validating user input, managing the risk may involve accepting it or transferring it. Insuring against the threat occurring or transferring the threat to another application to deal with (such as a Firewall) may be appropriate options for some business models.

The third point is that designers need to understand what they are securing, before they can appropriately specify security controls. It is all too easy to start specifying levels of security before understanding if the application actually needs it. Determining what the core information assets are is a key task in any web application design process. Security is almost always an overhead, either in cost or performance.

## What are Risks, Threats and Vulnerabilities?

Pronunciation Key

risk

(risk)

n.

1. The possibility of suffering harm or loss; danger.
2. A factor, thing, element, or course involving uncertain danger; a hazard: "the usual risks of the desert: rattlesnakes, the heat, and lack of water" (Frank Clancy).
3. a. The danger or probability of loss to an insurer.
   b. The amount that an insurance company stands to lose.

4. a. The variability of returns from an investment.

   b. The chance of nonpayment of a debt.

5. One considered with respect to the possibility of loss: a poor risk.

threat

n.

1. An expression of an intention to inflict pain, injury, evil, or punishment.
2. An indication of impending danger or harm.
3. One that is regarded as a possible danger; a menace.

vul-ner-a-ble

adj.

1. a. Susceptible to physical or emotional injury.

   b. Susceptible to attack: "We are vulnerable both by water and land, without either fleet or army" (Alexander Hamilton).

   c. Open to censure or criticism; assailable.

2. a. Liable to succumb, as to persuasion or temptation.

   b. Games. In a position to receive greater penalties or bonuses in a hand of bridge. In a rubber, used of the pair of players who score 100 points toward game.

An attacker (the "Threat") can exploit a Vulnerability (security bug in an application). Collectively this is a Risk.

## Measuring the Risk

While we firmly believe measuring risk is more art than science, it is nevertheless an important part of designing the overall security of a system. How many times have you been asked the question "Why should we spend X dollars on this?" Measuring risk generally takes either a qualitative or a quantitative approach.

A quantitative approach is usually more applicable in the realm of physical security or specific asset protection. Whichever approach is taken, however, a successful assessment of the risk is always dependent on asking the right questions. The process is only as good as its input.

A typical quantitative approach as described below can help analysts try to determine a dollar value of the assets (Asset Value or AV), associate a frequency rate (or Exposure Factor or EF) that the particular asset may be subjected to, and consequently determine a Single Loss Expectancy (SLE). From an Annualized Rate of Occurrence (ARO) you can determine the Annualized Loss Expectancy (ALE) of a particular asset and obtain a meaningful value for it.

Let's explain this in detail:

**AV x EF = SLE**

If our Asset Value is $1000 and our Exposure Factor (% of loss a realized threat could have on an asset) is 25% then we come out with the following figures:

**$1000 x 25% = $250**

So, our SLE is $250 per incident. To extrapolate that over a year we can apply another formula:

**SLE x ARO = ALE** (Annualized Loss Expectancy)

The ALE is the possibility of a specific threat taking place within a one-year time frame. You can define your own range, but for convenience sake let's say that the range is from 0.0 (never) to 1.0 (always). Working on this scale an ARO of 0.1 would indicate that the ARO value is once every ten years. So, going back to our formula, we have the following inputs:

**SLE ($250) x ARO (0.1) = $25 (ALE)**

Therefore, the cost to us on this particular asset per annum is $25. The benefits to us are obvious, we now have a tangible (or at the very least semi-tangible) cost to associate with protecting the asset. To protect the asset, we can put a safeguard in place up to the cost of $25 / annum.

Quantitative risk assessment is simple, eh? Well, sure, in theory, but actually coming up with those figures in the real world can be daunting and it does not naturally lend itself to software principles. The model described before

was also overly simplified. A more realistic technique might be to take a qualitative approach. Qualitative risk assessments don't produce values or definitive answers. They help a designer or analyst narrow down scenarios and document thoughts through a logical process. We all typically undertake quantitative analysis in our minds on a regular basis.

Typically questions may include:

- Do the threats come from external or internal parties?
- What would the impact be if the software is unavailable?
- What would be the impact if the system is compromised?
- Is it a financial loss or one of reputation?
- Would users actively look for bugs in the code to use to their advantage or can our licensing model prevent them from publishing them?
- What logging is required?
- What would the motivation be for people to try to break it (e.g. financial application for profit, marketing application for user database, etc.)

Tools such as the CERIAS CIRDB project (https://cirdb.cerias.purdue.edu/website) can significantly assist in the task of collecting good information incident related costs. The development of threat trees and workable security policies is a natural outgrowth of the above questions and should be developed for all critical systems.

Qualitative risk assessment is essentially not concerned with a monetary value but with scenarios of potential risks and ranking their potential to do harm. Qualitative risk assessments are subjective!

---