Abuser Stories

What do We Mean by an Abuser Story?

Most of us are by now familiar with the concept of a user story. User stories are a common practice that is used by most Agile or Scrum Teams to describe the features of an application. Developers will take the user stories and code them as features in their applications. An *Abuser Story* on the other hand is the "evil" version of a user story. Basically, an abuser story is a simple description of how the user story (feature) can be abused by a malicious actor.

Why Do We Need Abuser Stories?

Securing our applications is a highly involved and continuous activity. We don't want our systems compromised or our data to end up in the wrong hands. Ideally, we would like to ensure that security flaws and vulnerabilities are caught and addressed as early as possible. You should try to include at least one abuser story, where applicable, for every user story that you have. If done correctly, this can significantly reduce the number of security flaws being introduced into a feature thus ensuring that the application is much more resilient.

What Does an Abuser Story Look Like?

Writing abuser stories is an exercise in *"thinking like the enemy (hacker)."* It's a good way to help secure applications and stay ahead of potential attacks. If you're actively thinking of ways that your application may be compromised, then you're that much further ahead in securing your application. An abuser story is similar in structure to a user story. The typical structure looks like this:

As a ***<malicious actor>****, if I* ***<perform a malicious action>*** *it would result in* ***<outcome of malicious action>****.*

To switch to abuser story thinking, ask yourself:

- ✓ How can I interrogate the system for useful information by feeding it unexpected input?
- ✓ What happens if I feed it null or wrong input?
- ✓ How could I harm the system by calling a function repeatedly?

Examples can include:

URL Tweaking

As a malicious user, if I see what looks like my Member ID in the URL and change it, it will result in another user's account displaying.

XSS

As a malicious user, if I paste HTML that includes JavaScript into every field on the page, it will result in the execution of the pasted Javascript.

Authentication
As a malicious user, if I brute-force username and password fields, it will allow me to gain unlawful access to the application.

In Summary

Abuser stories are a very useful way to integrate security into your development lifecycle. Every time a feature is described someone should spend some time thinking about how that feature might be unintentionally misused or intentionally abused. By adding abuser stories to our “shift left” mentality we can ensure that our applications are safer from potential attacks.