* 1. **Pre-Application Walkthrough**
     1. **Burp Suite Setup -** Burp Suite Pro is a proxying tool - basically a mini-server we set up in between our computer and the application to capture requests that are sent by us to the application and the responses sent back. It's full of different tools for examining these requests and transforming them to see how the application will respond. NetSPI has basic settings that we always use with Burp Suite.
        1. S:\Assessment Team\NETSPI-U\burp\_netspistandard\_projectoptions.json
        2. Upload in to each Burp project – Aaron is this available in new Burp?
     2. This is the time to double check your proxy tool configurations, and make sure that no functionality is turned on that could potentially access parts of the application that we don't want to. Some links in a web page will kick off requests simply by navigating to them. On an administrative page, this could have catastrophic results. So what should be turned off?
        1. Content discovery/spidering! This is much harder to accidentally have on in the new version of Burp.
        2. Any other automated tools that you have installed! The only input that should be sent to the application at this point is **manually created.**

* 1. **Application Walkthrough - What's the point?**
     1. **Gather as much information as possible on the application, while looking for sensitive and/or vulnerable functionality in the application.** 
        1. If it's sensitive (think user account modification, password change, turning off the servers for half the company, etc), we don't touch it with automated tools.
        2. If it's vulnerable looking, we note it and come back later.
     2. **Try to answer essential questions about the application. Some examples are included below:**
        1. What kind of application is it? Do you have a basic understanding of it's function/use for the client?
        2. What kind of vulnerabilities could be present and majorly compromise the application? Can a bank account user look at another user's account? (Business Logic)
        3. Are there areas of sensitive functionality that shouldn't be touched by automated tools? What are some examples of functionality that shouldn't be touched by automated tools?
        4. What kind of software is the website using? (ASP.NET, NodeJS, Angular, etc)
        5. Is it behind a load balancer? (Look for an X-FORWARDED-BY header)
     3. **An application walkthrough is done manually the first time through, and can also be called User-Directed Spidering.** 
        1. More sophisticated and controlled technique of mapping a webpage/application. User clicks through a webpage and proxy/spider records results.
        2. Advantages:

More controlled/less danger of activating damaging admin functionality

User controls data submitted to the page

* + - 1. Disadvantage: Slow, you can miss things
      2. Hidden content can sometimes be found when spidering a web page. This is a gold mine for information on unreleased content, testing and debugging suites, config files, databsase credentials, source code, etc.
      3. Look at the robots.txt to see where the application developers don't want you to look.
  1. **Automated Tools: Content Discovery/Spider – Discovering and mapping with speed**

In some applications, directories could be "hidden", or not accessible through the GUI. Sometimes these hidden directories are not access controlled, and can hold valuable/sensitive info**. The goal of automated content discovery is to quickly discover all available areas of the application, so that we may quickly identify potentially sensitive areas for testing**. If you identified no areas of the application which would be sensitive to automated tools, or have a user account that doesn't have access to that sensitive functionality, then this tool should be run while authenticated under that user account. Burp Suite Pro has one large tool for content discovery.

a. The content discovery tool has two main pieces of functionality:

* + 1. Crawl to links discovered on pages and add them to the application map - **Spidering**
    2. Test pre-existing lists of common filenames and directories at each page, in order to search for hidden directories. -**Brute Force Automated Content Discovery**

b. Content discovery is available by navigating to the Target tab in Burp, right clicking the host you want to discover content on (MAKE SURE YOU SELECT THE RIGHT ONE), starting a content discovery session.

* + - 1. Once this is active, you can either keep the session you activated it from alive, or you can set the Scanner to update from the Cookie Jar at Project Options > Sessions > Cookie Jar.
  1. **Automated Scanning: WebInspect, Nessus, and Burp Scans**