Internal Penetration Testing: Attacking Systems That Matter

When you are conducting internal penetration tests in large environments, prioritizing attacks can be a challenging task, because of the number of systems and vulnerabilities. Attacks performed during testing are commonly prioritized based on the nature and severity of the vulnerabilities identified. However, the effectiveness of that approach can be greatly increased by focusing on the right systems. The goal of this blog entry is to share my thoughts on a few ways to identify those systems. Penetration tests typically have two primary objectives: find sensitive information and obtain Domain Admin privileges on the network. When you are trying to locate sensitive information on the network, it's important to put some thought into where data is commonly stored. Some of the most common locations include: email servers, TFTP servers, FTP servers, network file shares, and the almighty database server. All of those are good places to look for sensitive information, but I prefer to start with the database servers and work backwards. I also recommend targeting data stores after obtaining Domain Admin privileges, because Domain Admins usually have inherent access to the data. However, if the data stores are configured with weak or default passwords, it may make more sense to attack them first to ensure there is enough time to review the associated data. That approach is especially relevant to penetration tests that are conducted within the context of PCI or HIPAA. Now that we've talked a little bit about finding sensitive data on the network, let's talk about getting Domain Admin privileges. Obtaining Domain Admin access typically requires a little more effort, but there are some things that can be done to reduce the number of steps involved. Two of my favorites are attacking the domain controllers and systems with active Domain Admin sessions directly. Domain controllers can be attacked using traditional penetration testing methodologies, and attack vectors can vary greatly based on configuration. For that reason I won't be dedicating too much time to that specific area. However, if you're interested in learning more please refer to my previous blog: Windows Privilege Escalation Part 2: Domain Admin Privileges. However, Active Domain Admin sessions were not covered in any of my previous posts, so they may warrant some introduction. Active Domain Admin sessions are live interactive sessions created by a Domain Admin from one system to another. A simple example would be a Domain Admin logging into a member server from their workstation via remote desktop. When the Domain Admin logins in, they create a unique token that is stored locally on the member server. If a penetration test can gain access to the member server, they can potentially use the Domain Admin's token to impersonate them on the domain. So, if you can find an active session, you can potentially impersonate a Domain Admin. The question is, "What systems are the sessions active on?" Most penetration testers use a relatively straightforward approach that involves three steps:

1. Identify the domain controllers for the target domain.
2. Identify the domain users who are members of the Domain Admins group. This can be accomplished by querying the Domain Controllers for user information via RPC, LDAP, and in some cases SNMP.
3. Enumerate active Domain Admin sessions and associated systems. This can be accomplished by querying the Domain Controllers for a list of active sessions via RPC, or LDAP. Alternatively, this can be done by querying every system on the network individually for active Domain Admin sessions, but it takes a little bit longer.

There are a number of native Windows tools available for accomplishing steps 1 through 3. Most of

them come with standard Windows distributions, and the rest can be found in the resource kits for Windows 2000 and 2003. I suggest leveraging your favorite scripting language to streamline the process, but if you prefer to do it manually, feel free. Some people find the extra typing therapeutic. Whichever way you decide to do it, the following information should have been enumerated by the end of the process:

1. IP Address of the active session
2. Username of Domain Admin who started the active session
3. The start date/time of the active session
4. The idle time of the active session

Using this information you should be able to target systems that have the potential to provide Domain Admin privileges with very little effort. Whether your internal penetration test is driven by a client request, PCI/HIPAA requirements, or just an effort to better understand the threats in your environment, don't forget to save yourself some time by focusing your attacks on the systems that matter. Until next time, keep shining light into dark places. – SPS

Tool and Command References

- http://support.microsoft.com/kb/927229
- http://www.microsoft.com/Downloads/details.aspx?FamilyID=9d467a69-57ff-4ae7-96ee-b18c4790cffd
- http://technet.microsoft.com/en-us/sysinternals/default.aspx
- http://www.joeware.net/freetools/index.htm
- http://netspi.com/blog/2009/10/05/windows-privilege-escalation-part-1-local-administrator-privileges/
- http://netspi.com/blog/2009/10/05/windows-privilege-escalation-part-2-domain-admin-privileges/