EITS
Enterprise IT Security

RELEASE DATE:	CONTACT:
February 4, 2021	Mike Sullivan
	Sr. Security Architect
RELEASE METHOD:	Enterprise IT Security
Social Media and Website	info@eits.com

Correct Common Exploits to Prep for a Pen Test

Consider looking at security basics before adding another tool to the mix. Last week's introductory article on penetration testing discussed reasons to perform this type of security assessment. Regulatory compliance, testing defenses, and changes to the environment top the list. But a pen test does more than check a box for completed tasks; it gives you a picture of coverage and gaps in security. In essence, you have an idea of what you need to correct so you can avoid getting caught up in the "arms race" for the latest tools, which can sometimes turn out to be redundant.

If you have not participated in a pen test before, the third-party findings will likely contain some or all the relatively easy-to-fix problems I'm addressing here. By correcting these issues, yourself, you can "skip" that experience and remediate them before your next pen test. Each of the issues described leads to quick compromise or quick elevation of privilege, resulting in a complete takeover of the environment. In simplified terms, if your organization has any of these issues, then your security program investment is not providing you with an acceptable return because it can all be bypassed with old attack methodologies.

Default Insecure Microsoft Settings

Microsoft has a massive catalog of software and, more than likely, you are using a lot of it. All of this software interoperates out of the box and supports backwards compatibility by disabling some of the secure functionality. To compound this, pretty much all Information Security organizations are understaffed and overworked. So if you are not customizing your settings into an organizational baseline, it leaves gaping holes in security, only to be quietly exploited by attackers down the road. The most impactful example of this involves a trio of vulnerabilities that have been abused for at least 20 years:

Link Level Multicast Name Resolution – Link Level Multicast Name Resolution (LLMNR) is enabled by default on Windows operating systems, with the purpose of allowing communication between computers using only their name. While that may sound like an entirely useful and necessary function, when you understand how it works, you realize that it has no place in a modern business network; the feature was created for home networking where no Domain Name Resolution (DNS) server is present. If you have an Active Directory (AD) in your company, you have DNS servers. Even without Active Directory, if you have anything more than a flat network (meaning your network is separated into

multiple subnets), you have DNS servers. When LLMNR is enabled, it provides a last resort fallback for when a name cannot be resolved in any other manner.

> **Example**: You enable autocomplete in a browser (also a bad idea) and mis-key “goggle” instead of “google” You hit Enter, and because “goggle” does not exist in your autocomplete history, the browser does not append “.com” and instead looks in the local hosts file for a static entry for “goggle.” Because naturally this does not exist, the next step is to query the configured DNS server specified in the network settings. This is likely an internal DNS server which is configured to check its internal zones first for the existence of the name “goggle” and quite possibly append your domain name to the end of it. When that is not found, the DNS server queries the root servers on the Internet for the name “goggle.” When it is not found there, it falls back to LLMNR, which is a broadcast to anyone listening – essentially screaming “Who is goggle?”
>
> On a healthy network, nothing will respond unless there is a computer named “goggle,” in which case you probably already have a problem. An attacker exploits this problem by using software configured to respond to any broadcast request by replying “I’m <insert name here> and this is my IP.” Now, a typo has just inadvertently made you communicate with an attacker-owned computer and reveal information about yourself that allows them to impersonate you. It really is that quick and easy.

With LLMNR as the foundation of this trio, you probably noticed that it will only work on the local subnet and only if you “fat finger” a host name. The next building block that makes this worse is Web Proxy Auto Detect (WPAD).

Web Proxy Auto Detect –This feature is enabled by default on all browsers, but the problem specifically exposes itself in Microsoft browsers because Internet Explorer and Edge are highly ingrained for network communication. WPAD pre-dates Active Directory (AD); it allowed each workstation in a network to locate the web proxy server without manual configuration. On browser start, and frequently while the browser was still open, there would be name lookups for “wpad.” When this was used, there would be a server named “wpad” that would respond with a configuration file containing the IP, port, and options for the web proxy. The browser would then import this configuration, allowing the user Internet access. Since the creation of AD and the use of GPOs to configure machines, this is rarely ever done anymore, which created an opportunity for attackers to use the aforementioned attack methodology to force you to communicate with their system. Now an attacker does not need to wait for a typo, as a nonexistent name is looked up multiple times per minute.

To take it further: The LLMNR attack is still the same, which means the attack surface is still the local subnet. But what happens on the network when you plug in a new computer? Likely, when you request DHCP to get an address, you provide your hostname, and it is automatically placed in DNS for other computers to find you. What if you name your computer “wpad”? If you have Active Directory DNS, no problem; Microsoft already blocks that name. But with nearly any other DNS solution, “wpad” registers. Now *every* computer with a browser open is communicating with you. The attack surface just became *every computer with a browser open*, and the attacker is likely able to take over the domain in less than 5 minutes!

The third leg of this trio is Server Message Block (SMB) signing not being enabled by default:

Server Message Block –This is the perfect example of compatibility taking priority over security. The list of vendors who cannot support SMB signed sessions is dwindling and becoming less of an issue, which offloads the problem to security professionals. They must now ensure the latest software versions are being used so that this feature can be enabled. A regular unsigned SMB communication consists of authentication and commands. The LLMNR attack allows the attacker to relay or replay the authentication they receive from you when you inadvertently communicated with them; the attacker adds the commands, and the impersonation occurs. SMB signing adds a signature to the SMB communication. Skipping the details of how public key infrastructure works, the signature portion of the communication is an encrypted blob that can be decrypted by the destination machine and compared to the source information from the request origin. If the source and the signature match, then the authentication may proceed and commands may be run; if the source and the signature do not match, then the request is thrown away.

If you've been paying attention, you probably understand that SMB signing alone can prevent the previously described attack; however, the reason I broke this down into three pieces is because enabling SMB signing won't prevent other possible impersonation techniques when LLMNR is enabled. *Disabling LLMNR is by far the easiest and fastest way to prevent a multitude of attack types*, and it should have zero impact on your network. *Disabling WPAD* in the browsers should equally have no impact to your environment. In addition, if you are using a third-party DNS solution, create a static record for "wpad" and assign it the address "127.0.0.1" to prevent any attacker device from being able to register that name. SMB signing will require some protocol discovery or, at a minimum, a technology inventory to understand if you have systems that will fail when the third portion of the SMB communication is used.

Incorrect Group Policy Object Permissions

GPOs simplify computer management within Active Directory (AD) and are often used to set standards or baselines across an entire organization. They really make a system administrator's life easier; however, there seems to be real confusion for many revolving around how to apply the settings in GPO to a group of users or computers, which results in huge holes that attackers may leverage to take over an AD domain.

All GPOs must be linked to an Organizational Unit (OU) to be applicable to those users or computers. This seems to be the easy part and is well understood; after all, troubleshooting this is relatively easy. If the GPO is not linked, it just does not work. However, we see the "Delegation" tab in a GPO consistently misused, and there is no warning indicator when this occurs. The GPO is applied as expected and the admin moves on.

By default, a GPO's Delegation properties contain READ access for the Authenticated Users group and EDIT access for the Domain Admins, Enterprise Admins group, and SYSTEM. This should make perfect sense, as Domain Admins are typically the ones to create or modify GPOs, while members of the Authenticated Users group, which contain both user and computer objects, just need to read the GPO to know which settings are to be applied.

It should also be understood that every setting in a GPO is essentially a command and will be run as the permission level of the user or computer to which it is applied. From an attacker's perspective, this means code execution. EITS commonly sees modification to this default configuration by confused admins who add the Domain Users group and FULL control; this inadvertently allows anyone to modify the GPO, which may allow an attacker to lower the security level of the target or add new settings that can execute code. Depending on the OU to which the GPO is linked, this can have far reaching effects, such as the Default Domain Policy GPO.

An exacerbating problem is that logging for GPO modification is not enabled by default and, even when enabled, does not provide a list of changes made to the GPO; this makes it impossible for anyone to know when the change was made or by whom. Customers estimate that many of the misconfigurations discovered during penetration testing engagements are multiple years old. When attackers discover that, they can take over a domain in less than 5 minutes!

Rarely should the Delegation tab on GPOs be changed from the default, unless your organization has decided to further restrict GPO management to a group smaller than Domain Admins. In that case, the Delegation tab should still look the same across all GPOs; check them regularly for inconsistencies. If your organization follows a change-management process, then enabling the events associated with GPO changes helps to determine if modifications are being made outside of change windows or by users who are not permitted.

Server Out-of-Band Management Interfaces Enabled

Many physical servers ship with a dedicated Network Interface Card (NIC) for out-of-band management. When connecting to a server on this interface, you receive a Graphical User Interface (GUI) for server management that contains low-level access to the physical hardware such as power, disks, and sometimes full-featured console access. Each vendor has its own name for these interfaces, such as iLO, DRAC and IPMI. The problem is that the software for these interfaces is always old and outdated, containing multiple vulnerabilities that result in admin level access.

All an attacker needs to do is get on the internal network and look for these web interfaces; when they are found, the attacker exploits them to gain access to the management software. At this point, there is a good chance that they will then be able to gain access to the console of a machine where an admin is still logged in from the last time troubleshooting the physical hardware.

The impact here is hit or miss, depending on whether your admins are trained to log out every time they leave a server. Without a console session, usually an attacker can only create a very annoying Denial of Service (DoS); however, what we often find is that customers are not using the interface or did not even know it was active. When servers are racked in a datacenter, it's not always the System Admins who are doing it. The deployment team or the datacenter team do not get to deal with the pain of vulnerabilities, but merely plug cables into all interfaces as instructed. The fix here is easy: If you are not using these out-of-band interfaces, do not plug them in. Consider adding a step to your deployment process where these interfaces are tapped over to ensure they are never accidentally connected.

Overly Permissive Antivirus Exclusions

Our final common finding on penetration tests is Antivirus exclusions. We all understand the importance of Antivirus and generally understand how it does its job; however, troubleshooting Antivirus problems is not fun in an enterprise setting. This results in applying exclusions that are much too broad. The user is happy, the admin is happy, and everything resumes as normal; the problem is that inadvertent holes that do not set off alarm bells have been opened, much like the other issues in this article. Networking theory promotes the concept of **creating the tightest rule possible** when it comes to firewalls; unfortunately, most Antivirus administrators do not have this training and do not follow this concept.

Most Antivirus solutions store the list of exclusions in a clear text format, readable by an unprivileged user who knows where to look. We have seen them in log files, registry keys, and sometimes the GUI itself. Do not consider the excluded locations to be a secret. Following that understanding, there are some common rules to follow when creating exclusions:

> **Never create a file extension exclusion without a full path**. Doing so provides an attacker the ability to execute malicious code from anywhere on the machine just using that extension. Attackers are not restricted by file type associations; as such, malicious code can be loaded onto a machine and run with any file extension.
> **Specify exclusions by filename or file type**. It is common to see entire folders excluded, and this is sometimes necessary. Be sure to scope these exclusions down as tight as possible. In reality, the correct answer is going to be to specify the filenames or file types within that directory rather than the entire thing.
> **Check destination folder permissions for excluded objects**. The final and most significant problem with Antivirus exclusions is excluding files or folders in user-modifiable locations. If you exclude anything within the user profile directory, an attacker will be able to leverage it, either by placing the malware in that folder or by renaming the existing excluded file and replacing with the malware. The problem is not limited to just the user profile directory, so check the permissions for the destination of any excluded object.

Conclusion

If your team does not conduct penetration testing, develop a plan to systematically challenge your defenses, preferably with help from third-party professionals who can guide you through the process. Even if your organization performs routine testing, time frames are highly compressed, so not all testers will look for these older problems – you may still be vulnerable to classic exploits even though previous pen tests did not flag them. Remediating these vulnerabilities in advance will allow you to avoid findings or devote time to more problematic ones.

So before adding new bells and whistles, make time to inspect these simple configurations, harden your environment, and prevent findings on your next pen test. In the next installment about penetration testing, I will describe some of the tests performed and what to do with your findings.

EITS
Enterprise IT Security

The experts at EITS focus on helping industry professionals develop robust security practices. Please share with any peers and partner organizations you think may benefit from this – sharing information improves everyone's security posture.