Configuration Security: Identifying & Avoiding Mistakes

This article will not cover explicit configuration settings but will cover the ways to determine if the configuration settings are appropriate and potentially assist with identifying configuration mistakes that can lead to information leaks, automated attack successes (password spraying, worms, etc.). First there will be clarity provided about what constitutes a configuration, an industry best practice configuration, an organization best practice configuration, an unsecured configuration, a configuration mistake, and a compensating control. Once the meanings are settled baseline configurations will be covered and will include tailoring. Configuration management will follow with a wrap-up discussion about compensating controls.

It is always best to ensure all parties involved have the same understanding of terms they will be using so this sections defines the way in which this article defines the following terms: configuration, industry best practice configuration, organization best practice configuration, unsecured configuration, and configuration mistake. A configuration is any setting that can be manipulated to enable, disable, or manipulate any feature or functionality in hardware, firmware, or software. Simplified, it is a setting that can be changed with the example of “on”, “off”, or “low power mode”. An industry best practice configuration is a setting that has been determined by an industry vertical to provide sufficient security to prevent the majority of threats. Basically, it is the industry agreed setting that is most likely strong enough to stop most hackers with an example being 8-character complex passwords. An organization best practice configuration is a tailored industry best practice configuration that aligns with business needs and goals. In layman’s terms, it is a business specific setting that meets or exceeds industry best practices but which has been changed to enable business users. The provided example being the use of QR code badges that are ‘linked’ to 22 character complex passwords with 22 character complex passwords exceeding the industry best practice. An unsecured configuration is any configuration that does not meet industry best practice configurations with an example of an administrative account using the default account name. A configuration mistake is any unsecured configuration that also has no compensating controls and example being an administrative account that uses the default account name and which does not limit which remote hosts or local personnel can access the device, software, or firmware. A compensating control is a security mechanism or mechanisms by which the risk of an unsecured configuration can be reduced to an acceptable risk level, which is usually the risk level achieved had the unsecured configuration been secured with an industry best practice. An example of a compensating control that continues the admin account with the default account name example, may be the use of a firewall to limit the remote connections to a single jump-box, which has tightly controlled user account ACLs, while having the physical device locked in a secured cage, to which only two people have keys, in the server room to which has logged and audited badge access. The simplest way to secure everything without going through every setting it to create baseline configurations.

There have been numerous times then I have gone through as many configuration settings as I could find on an operating system to include some windows registry settings such as software write blocking and Linux kernel file settings such as preboot passwords. The number of configuration settings are ridiculous and more than once I created settings that made my systems unusable. Thank goodness for baseline configurations and the ability to automate the process of configuring the most common firmware and software. There are a number of resources for obtaining baseline configurations that can be easily found using a quick internet search so we shall skip ahead to what happens after the industry baseline configurations are obtained. Baseline configurations are such a time saver but may inhibit workflows in an organization so they need to be tailored before they are deployed throughout the network. There are differences of opinion about the scope of tailoring, such as if they should be tailored to be department specific or if an overall configuration baseline that supports every department inclusively is sufficient. The truth is both have merits and it is organizational dependent. Tailoring to the department level is much more granular and provides for greater security but also comes with higher complexity and administrative cost while conversely, tailoring to the organization level may be better for small organizations that can not fund the increased administrative cost and have little security variance needs between departments. Whichever solution is chosen it should start with an industry accepted security baseline. The baseline should be placed on a couple of 'test' users with detailed reporting of any configurations that need to be adjusted to permit the completion of business specific work tasks. Once all the configurations have been adjusted the tailored settings are recorded as the enterprise configuration baseline. A review of all configuration changes is required to ensure any unsecured configurations have been supported with appropriate compensating controls and that those specific compensating controls are annotated in the enterprise configuration best practice documentation. Once the documentation is complete the enterprise can begin a thoughtful and careful deployment of its configuration best practice baseline.

As businesses are dynamic and have changing needs there will be times when configurations will need to be updated to support the new business requirements and needs or be deemed to restrictive and an inhibitor to successful completion of work tasks. This is one third of configuration management. Configuration management is a process by which configuration that have been deployed as part of the enterprise configuration baseline are monitored for changes, inappropriate changes reverted, and legitimate changes are approved and deployed. There are tools that automatically check configurations that in used to determine if they are aligned with the deployed baselines, some of these tools can even be configured to automatically revert any inconsistencies. When the time comes for a configuration to be updated or changed due to changing business needs there should be a well defined request and approval process that includes thoroughly documenting the requested change, the business reason or justification for the change, the compensating controls that will be deployed to promote sustained security after the change, how the change will be tested to ensure the compensating controls are effectively working, with an updated version to the enterprise security configurations best practices documentation at a minimum. It is encouraged to seek more complete literature on configuration management than can be provided in this article as it is an essential part of any security program.

A critical part of the tailoring and updating of configuration settings is the use and implementation of compensating controls. Any secured configuration that is mad unsecured needs to have compensating controls or it will be a configuration mistake. Likewise, any unsecured configuration that is not secured and has no compensating controls is a configuration mistake. They are both mistakes even if the risk is accepted because they increase the attack surface with no attempts at mitigation. It is not uncommon for an organization to make or leave a configuration unsecured, implement a single compensating control, and assume the same level of protection has been achieved. While some level of risk has been reduced it is quire infrequent that a single compensating control can achieve the same level of risk mitigation that a best-practice control achieves. It is more often the case that two or more compensating controls are required to achieve similar levels of risk mitigation which makes the use of compensating controls very administratively taxing while substantially increasing the complexity required to achieve security. In the very simple example of the administrative account that had the default account name the compensating controls were to create a jump-box, restrict access to the jump-box, create restrictive accounts on the jump-box, update firewall rules to only permit access from the jump-box to the platform that hosts the administrative account with the default account name, and ensure the device with that administrative account was locked in a location with highly limited and audited access. All of those compensating controls because the built-in administrative account could not have it's account name changed or be disabled. This scenario is actually fairly common and there are thousands of configurations on a standard OS industry security configuration baseline. The key to compensating controls is to implement all of the configuration best practices unless there is a critical need to deviate from the best practice. Compensating controls are a very complex and administratively burdensome configuration requirement so determining whether changing a business process or adding compensating controls is most cost-effective may not be as transparent or easily decided as may first be thought which is why configuration management and requested changes need to scrutinized from a business process and administrative strain perspective.

This article did not cover explicit configuration settings but did cover the ways to determine if the configuration settings are appropriate and potentially assist with identifying configuration mistakes that can lead to information leaks, automated attack successes (password spraying, worms, etc.). there was clarity provided about what constitutes a configuration, an industry best practice configuration, an organization best practice configuration, an unsecured configuration, a configuration mistake, and a compensating control. Once the meanings were settled baseline configurations were covered and included tailoring. Configuration management followed with a wrap-up discussion about compensating controls.

Thank you for reading

-Andrew Kosakowski

www.Anthko.com