Cybersecurity Basics: 12 Basic Security Components

It is quite often that I hear security professionals state organizations need to return to the basics. There are even times where security professionals have amassed various lists of basics when it comes to a cybersecurity program but (in my opinion) many of the lists are incomplete or focused on a certain aspect of a security program (the software development team for instance). In this article I will attempt to pull together a holistic list of 12 basic requirements to have a minimally functional security program. I will not focus on specific technologies just as intrusion prevention systems, firewalls, and endpoint detection and response as the scope of this is focused on the strategic program level and not the architectural implementation of that strategy. Although we will not be going into architectural specific detail we will be covering some of the concepts and policies any selected architecture should be able to provide. For those whom want a "quick list" of the topics I will be covering I have added one here:

1. Account Management
2. Asset Management
3. Asset Security
4. Patch Management
5. Encryption Management
6. Logging and Auditing
7. Compliance Management
8. Configuration Management
9. Security Awareness
10. Incident Response
11. Acquisition Security
12. Insider Threat and Personnel Security

Accounts are critical to every business as they permit users, services, and scripts access to resources to meet business objectives but securing accounts continues to prove be a lacking. It seems nearly daily that I hear a news story about accounts being compromised using a password attack, phishing, hash grabbing, or any number of other techniques. When it comes to account management we must consider the life-cycle of each account type and develop plans, policies, and guidelines for each essential account type and bar any other account type. For user accounts, the life-cycle usually consists of request for creation, creation, password issuance, account life, password resets, breach remediation (plan for this), notification of decommission, decommission, retention and archival of username. The request for account creation must follow a secure process to ensure legitimacy, confidentiality, and non-repudiation. The account creation process must be highly restricted to very few personnel with strong logging and auditing. Password resets should also be highly restricted with embedded user verification processes. Account password reset questions should never be historical based, have a limited number of possible answers (ex. what make was your first car,) or be something that is easily obtained from internal employees or external entities (ex. who is your supervisor). Breach remediation should be based on your business processes, the criticality of the account, your incident response maturity, it should not be based on the complexity of implemented technology. Many organizations forget and neglect the next two steps so it is critical to ensure your organization develops plans, policies, and guidelines with enforcement for notification of deactivation and deactivation. Managers should be responsible for notifying the IT or Security department when a user account is no longer needed or in use. This notification can be part of the HR exit process, a separate manager ticket submission process, or a different business defined process. Once the notification is sent the IT or Security department must have a tool that lists all accounts that user is associated with or which that user was responsible in order to prepare those accounts for transition or deactivation as well. When it comes time to actually deactivate an account administrators should schedule the deletion (unless business need requires immediate deactivation) to run as part of a script. The script should run daily have two parts: the first part should deactivate all users that have been identified by the administrator; the second part should deactivate all users that have not been active within a defined time period which will often be set by compliance standards. Service accounts that are embedded in software and hardware should be disabled if not required while having names and passwords changed if they are required. Service accounts that are created to support interoperability or projects should have passwords changed regularly (ex. every 90, 180, or 365 days) due to hash grabbing and other password attacks. They should also follow similar creation and disablement procedures as user accounts. Scripts (and scanners) that require a password to perform a function should never have passwords stored within the script themselves. The passwords they use should be changed every 30 days.

Asset security and management are often constrained to the definition of assets owned by the organization, for our purposes we will expand this definition to all devices and software that are located on the corporate network to include authorized and unauthorized devices as well as personally owned and corporate devices. Assets come and assets go (sometimes without permission) and on occasion there are even unauthorized assets so it is imperative you know what you own, where it is, and when it should be sunset or refreshed. An asset can be any hardware, software, or firmware that is integrated into the enterprise environment. The first stage of any asset is acquisition. Once an asset has been acquired it should be inventoried. BYOD devices are employee or stakeholder owned assets but should still follow an inventorying process which is usually synonymous with the enterprise authorization process. For non-BYOD devices enterprises should create three disparate types of inventories: Physical assets, digital assets, and licenses. Physical assets should be tagged with detailed identification information indexed in the asset management database. The indexed information should consist of, at a minimum, MAC address if applicable, serial number, manufacturer, model, date acquired, date of sunset, and location (or employee) to which it has been issued. The digital asset inventory should include vendor, software version, date of acquisition, date of sunset, and issuing information. Licensing inventories should include acquisition dates, renewal dates, total licenses purchased, number of licenses used, where the licenses are in use, and personnel whom are authorized to use the licenses if applicable. With these three databases created and updated to reflect changes in issuance or acquisitions finding rogue devices and software should be significantly easier to identify. Any asset that is identified on the network that does not correspond to a device in the corresponding database should trigger an alert to a security professional whom can identify the reason. Identification an be performed using a scanner that drops the output to a file with an automated script that compares the scan results to the databases or a security device that checks network traffic against the inventory databases. This technique can also alert on devices that are passed their sunset dates (date of planned decommission). Sunset dates should hard dates that have corresponding strategic plans and budgetary line items. Hardware assets that are sunset should follow secure sanitation or destruction processes and if outsourced should be supported by strongly worded and ensured SLAs.

Patch Management is an area that has caused many heated debates about the need to balance support for legacy functionality while ensuring the latest threats are effectively mitigated. This article will not solve all of the patching heartache but is designed to reinforce basic quality practices that ensure patches are securely deployed in a timely manner after they have been tested for functionality and unintended consequences. The patching cycle typically will follow the patch-retrieval, test, acceptance, partial deployment, full deployment, and if necessary, roll-back processes. A key precursor to an effective patch cycle is having a test environment that mirrors the production environment (with a lack of pii, never have pii outside the production environment). A test environment is highly valuable because it permits the deployment of patches to determine if they have any obvious critical issues that would have detrimental impacts on a production environment. Patches need to be retrieved from a vendor and moved around the environment for testing and deployment. All this movement needs to be secured in order to ensure patches are not corrupted in transit and to ensure no rogue patches are added to the transmission. The best way to perform these security actions is to ensure patches are hashed and signed and that all transmissions are secured with encryption. Any patches left "laying around" in a database, on a hard drive, or in any other storage location should be encrypted at rest. During the testing phase, a designated user(s) should run a series of predefined steps that mimic normal production activity and report any anomalous results. The reported anomalies should be reviewed to determine if further testing is needed to evaluate any possible unintended consequences of the patch, for instance and OS update that disables or corrupts a network driver. Once the patch has passed the test process and the report has been accepted as being within the risk threshold the patch(es) should be pushed to a small group of users in production for a predetermined amount of time (usually one week) in an effort to identify any issues that were not experienced or observed in the testing phase. Once all risks have been identified and accepted, rejected, or remediated the patch should be pushed to the rest of the environment.

Encryption management is a rather complex subject that includes encryption at rest and encryption in transit for the purposes of this article encryption for data in use is not considered a basic security component. Encryption at rest is no longer restricted to data that is stored on servers and workstation but has expanded to encryption of data in the cloud, data on mobile devices, encryption of removable media, encryption of data in peripherals such as printers, and encryption of data in smart (ex. smart boards, smart TVs, personal assistants, etc.) and IOT devices. The key to encryption is encryption keys, meaning, encryption keys must be regularly rotated because modern threats are capable of using a variety of attacks in attempts to compromise keys. Some of the attacks are a bit well known such as virtual memory key recovery while others may still have more limited public knowledge. In addition to being rotated regularly, keys should also be fairly long and complex. The type of encryption used should be commensurate with the latest acceptable standards and encryption algorithms is use should have developed plans for decommission and replacement as it is possible that an algorithm will become obsolete by the revelation of being broken or technology having advanced to the point where the algorithm no longer provides sufficient strength. My personal favorite algorithm is serpent but it isn't widely used so used the algorithm that best suites your business needs and meets acceptable strength standards. For encryption at rest, entire drives should be encrypted. Where possible, technical security enforcement should be used to ensure no data can be moved to a device that is not currently supporting full disk encryption with an authorized algorithm. For encryption is transit, always use the strongest encryption possible and disable any compatibility with weak or broken algorithms or cipher-suites to prevent down-grade, stripping, and offline attacks. Always ensure you are enforcing HTTP with HSTS.

As logs are created they should actively be monitored for indicators of active threats and attacks for a predetermined time, then the logs should be retained for a in an archive for a defined period, during the archival stage the logs should periodically be scanned for indicators of previous compromise. The key to active log monitoring is to determine which logs are the most relevant and insightful without collecting all logs from every source because all logs are not created equal. While there are a substantial number of resources that detail which logs are best to review for each piece of technology it is essential to understand there should be an enterprise review to determine if any enterprise specific need require additional logging. Essential active logs should always be sent to a secondary on-network device and, if applicable, replicated to a remote off-network location. These logs should be automatically audited using security devices such as SOARs and SIEMs. These tools can be used to identify threats and attacks that correspond to an externally created rule or signature, anomalous activity, as well as internally developed indicators. When a threat or attack is identified these devices (most of them) can take automated remediation action if configured accordingly. Active log auditing is a powerful near-real time and recent history tool but are not designed for review of long-term archived logs. The typical deviation between what constitutes recent history logging and archived logs is typically ninety days. After logs have been within the auditing database for ninety days they are typically transitioned to the archival database for compression, encryption, and long-term storage. Long-term storage is often defined as twelve months from the date of the origination of the original log creation date. This can be subject to regulatory requirements that require longer retention periods. My personal preference is a fifteen month retention period which is twelve months after the archival date but archival log retention periods must be defined by organizational policy. After logs have been archived there will be periodic need to retrieve the archives with the goal of scanning them for indicators of past compromise. For the purpose of this article indicators of past compromise are defined as newly publicized indicators of compromise that detail the TTPs of threat actors that have been caught in the wild. When archived logs are are scanned for past indicators of compromise the goals is due diligence in determining if the organization was compromised through a security gap that was not previously identified and remediated. After the archived logs are scanned for past indicators of compromise they are returned to the archival database or securely deleted depending on the methods used to scan them. If they are returned, it is essential that the archived logs be hashed prior to being accessed, prior to scanning, after scanning, and after being returned to the archive. Active log auditing and archived log auditing are essential steps in maturing every security program and one of my personal favorite three security maturity steps.

Compliance should never be the reason to implement a security solution but every security solution implemented should be mapped to all compliance requirements that it meets. The goal of security should always be the protection of corporate personnel and assets. There have been many times when I have asked an organization how they met a specific compliance requirement and they stated that they did not meet it in any way because they had not mapped their existing solutions to compliance requirements. Often time there is also this notion in compliance that a requirement must be met in a very specific way that is often times at odds with the business needs and practices of an organization. While that is occasionally the case, often times there are a number of ways to meet a compliance requirement that aligns with a business need or practice. Ensuring compliance fulfillment does not come at the cost of business is a very delicate negotiation as some business practices are inherently risky. Many organizations have more than once compliance requirement as well which further substantiates the need to have a unified mapping of compliance requirement fulfillment to acquisitions.

Configuration management has been discussed at length by a plethora of resources so we will briefly cover the high-level needs that should be considered when implementing this component of a security program. Every piece of hardware, software, and firmware has configurations that can increase or decrease functionality and security is not only one of those functionalities it is also a consequence of configurations of other functionalities. When an asset is approved or acquired it should be analyzed to determine possible configuration options then compared against the needs of the business. All security configurations should be placed in the most secure-least restrictive mode, meaning, they should be configured to be secure as possible without inhibiting official corporate business. All configurations that correspond to functionality that does not map to a specific official corporate business need should be restricted and disabled. There are security configuration checklists for most common operating systems and devices that greatly increase the speed of these reviews and which are freely available. Once the configuration or implemented the ability to change those configurations should be highly restrictive and only permitted by a small number of administrators with strong review and approval processes for any request for configuration changes. Technologies (like scanners, network mappers, host agents, etc.) should be used to ensure configurations are not changed without authorization. If a configuration is changed without authorization it should 'automatically' changed back to an approved state.

With the number of social engineering techniques that are becoming increasingly successful against organization, security awareness is reflectively becoming increasingly critical. Now, I use the term "security awareness" and not "security awareness training" for a very particular and nuanced purpose. Training is mundane and employees don't want it, they will rush through it, and most will assume they remember everything from the previous cycle. Security awareness needs to be embedded in the culture of an organization and not a function of compliance goals. Creative campaigns need to be deployed to keep employees engaged while imparting much needed security knowledge. There has been a push for gamification of security training which has its merits but also has its specific target audience. When creating security awareness there should be a set of goals that are being sought with quality metrics that will assist in evaluating the achievement of that goal. For instance, let's assume that a company has been experiencing an increased number of phishing emails, has informed their employees of the situation, and employees are reporting a great number of spam emails as phishing emails because they are not clear on the difference. The organization could create an inter-departmental challenge to determine whom can report the greatest number of phishing emails the following month with the least amount of spam emails being reported. The organization could create a small (less than 10) list of rules for each department with the winning department receiving a reward, maybe leaving work two hours early on a Friday. This competition may (depending on culture) promote employee engagement, security awareness around phishing, and reduce the workload of anyone whom might be triaging reported suspected phishing emails. Security awareness should be culture specific with the goal of employee engagement and education while reducing or limiting the impacts of fear/false reporting of non-incidents.

Incident response should be determined by an organizational strategy that has been created prior to an incident. The strategy must first identify if there is a differentiation is response to assets of differing criticality, for instance, does a authentication database compromise follow the same overall response goal as a guest laptop? Once the criticality tiers have been created organizations should determine the desired outcome for incidents, (IE. should they be left in production and remediated while running, prepared for possible legal action, wiped and imaged, etc.). For some organizations this strategy may be to call an outside entity to investigate and remediate the issue while others may have internal teams that are dedicated to the resolution of any confirmed incidents. When it comes to internal incident responders, guides should be built for routine incidents but greater autonomy of action should be permitted for less routine incidents. There are incidents that appear to be a typical common-cause incident that unfold to be a rather advanced intrusion so responders should be given the flexibility to pursue the investigation until their skill level is expired and the situation needs to be passed to a more skilled practitioner. Communication is key in all area of security but when it comes to incident response there needs to be special care taken to ensure there is a clear communication path from an initial responder to the executive team (no a direct link from admin to executive in most circumstances though) so as to let the executive team mobilize PR, legal, and other assets as needed.

Acquisition security for the purposes of this document relates to having redundant supply chains with SLAs that guarantee priority of service in the event of a shortage or rush as well as ensuring the vendors that are used have a low risk of going out of business. When it comes to redundant supply chains (as shown by recent history) it is essential to ensure the secondary, tertiary, and so forth supply chains are geographically dispersed enough to overcome any regional issues in production or shipping. SLAs must also cover quality of receivables as well as transportation security methods. There is no reason to purchase equipment if a delivery driver manipulates the order, steals, or the product is shipped damaged, incomplete or ineffective.

Insider threat and personnel security are two inseparable components of a security program. Insiders do create real damage to corporations through bringing work content with them when they depart, intentional destruction due to hostilities, covert exfiltration due to bribery or blackmail, for the purposes of exposing alleged corporate wrong doing and the such. Prospective employees should receive a background check to determine if any legal action has been successfully taken against them that directly relates to the position for which they are being considered. If something is discovered, a risk assessment should be conducted to determine if the discovered item presents any significant risk or if it is not associated with the role and responsibilities proposed or if it was far enough back in time to be deemed inconsequential. Once prospective employees are hired their should be an anonymous insider threat reporting infrastructure that supports employees whom believe something is reportable but do not want to be identified as the reporter. The insider threat reporting infrastructure should be well-known in the organization with access, or information about how to access, easily identifiable, private, and highly accessible. There should be internally developed processes with HR and managers on how to act upon reports of suspicion.

Thank you for reading

-Andrew Kosakowski

www.Anthko.com