Week 5 Final Assignment—Application Security and Cryptographic Algorithms

Alicia Piavis

CST316: Information Security Management

Amr Elchouemi

6/7/2020

Week 5 Final Assignment—Application Security and Cryptographic Algorithms

One of the ways that organizations can support CIANA (confidentiality, integrity, availability, non-repudiation, and authentication) and promote security awareness is through the implementation of organizational security policies and procedures. Some of the components that organizations might include in these policies and procedures include an acceptable use policy, a bring your own device (BYOD) policy, a password policy, risk management (RM) procedures, a security awareness policy, a security enforcement policy, authorization and authentication policies, governance policies, and a privacy policy. While these are all very important components to include in an organization's security policies and procedures, some of the easiest to implement that also provide a high return on investment include acceptable use policies, password policies, security awareness policies, and authorization and authentication policies.

Acceptable use policies contain guidelines concerning how company hardware and software should be used in order to mitigate security threats. These policies are important so that employees have clarity regarding how they should and should not use company devices, as well as the company network. Improper use of devices, such as downloading software without approval, or surfing undesirable websites can introduce the organization to vulnerabilities such as viruses, malware, and even social engineering. The implementation of acceptable use policies supports basic security principles. For example, blocking certain websites on the company network can support defense in depth, creating a layer of protection by preventing access to risky websites (Sveikauskas, 2018). Another policy that is fairly easy to implement is a password policy. Password policies are important because they establish password requirements and regulate how often passwords should be reset. Password complexity is very important, as it can influence the ease with which attackers can crack a password and gain access to a system. By

requiring that passwords be a certain length, and that they have letters, special characters, and numbers, an organization can reduce the probability of an attacker gaining access to the system. Password policies also support defense in depth, as well as the principle of establishing secure defaults (Sveikauskas, 2018). By requiring that passwords meet certain criteria, organizations are able to set a default, or baseline level of security.

In addition to acceptable use and password policies, security awareness policies are important in educating employees about information security principles, how to avoid opening suspicious emails and clicking on malicious links, and how to respond when a threat is suspected. By instituting mandated information security training, the security principle "don't trust services" is supported (Sveikauskas, 2018). Don't trust services means educating individuals regarding which types of emails, links, and websites can be trusted. By making employees aware of security threats, organizations can reduce risk further. Finally, authentication and authorization policies should be established. Authentication refers to the verification that occurs to ensure that a user is who they say they are, and authorization refers to providing specific levels of access to a particular identity. These policies support two security principles, defense in depth and the principle of least privilege. By utilizing multifactor authentication, organizations can make the security of a system more robust and harder to penetrate. In addition, access control policies establish who has access to what, and limits an individual's permissions to that which they need to carry out their job tasks. By limiting the amount of information available to individuals, risk is reduced.

Together, acceptable use policies, password policies, security awareness policies, and authorization and authentication policies can be integrated with other security practices to establish the security of an organization. For example, the organization that I currently work for

implements an acceptable use policy that prohibits employees from accessing social media websites or personal email accounts while on the company network. This not only keeps employees more productive, but also prevents unnecessary risks from being introduced to the company network through malicious emails, social engineering, and viruses acquired through risky websites and links. In addition, the password policy at the organization requires that all network passwords are at least eight characters, and contain at least one number, one letter, and one special character. This increases the complexity of passwords, reducing the chance of an attacker cracking a password and gaining access to the system. All network passwords must be reset every 45 days. The security awareness policy requires that new employees receive information security training within 30 days of onboarding. The training includes topics such as phishing, social engineering, malicious links, locking devices when unattended, recognizing suspicious emails, and what to do when a threat is suspected. Finally, multi-factor authentication is used to make it more difficult to access the company network. In order to gain access to the company intranet, users must log in with their company email and password, accept a message sent to their cell phone from Okta, an identity management company, and then they are redirected through the browser to the company's homepage on the intranet. In order to implement authorization, the organization utilizes Microsoft Active Directory to add or remove users from groups, and add or remove permissions from groups. This allows the organization to easily track who has access to what, and limit the information that employees have access to.

In addition to these policies, network security mechanisms should be implemented to prevent and detect possible attacks. For example, dynamic host configuration protocol (DHCP) can be used to enforce whether or not a device can connect to a network, based on rules that an administrator sets (Wills, 2019). The organization can whitelist suspicious email addresses and

IP addresses, so that the IP addresses of login attempts are cross-referenced with a list of acceptable IP addresses that can gain access to the network. Internet Protocol Security (IPsec) should be leveraged to send data securely over the network (Wills, 2019). Access control should be used to grant or deny access to the system, grant or deny access to objects within the system, and track information regarding what actions were performed against objects by a specific subject. Role-based access control is recommended, as it is a common form of access control used today, and is based on assigning permissions to specific roles, which are then assigned to individuals. Email filters should be used to detect suspicious emails and send them to a spam folder or other directory. Multifactor authentication (MFA) should be implemented to increase the security of account logins: "the FBI recommends MFA that utilizes biometric information or behavioral information, such as geolocation data or internet protocol (IP) address" (Sargent, 2020). Additional security layers should include firewalls and network alerts for abnormal activity (Sargent, 2020).

Vulnerability scans and audits should be implemented to identify potential vulnerabilities in the network and across company devices. As Orrill (2020) states, "Vulnerability scanners can help an enterprise's IT staff identify weaknesses throughout its network, such as ports that could be accessed by unauthorized users and software lacking the latest security patches, helping to ensure network compliance with the organization's security policy". Active scanning should be used to simulate attacks and detect weak nodes based on the responses received from transmitting data. Active scanning should also be used to repair detected weaknesses. Some examples of network scanning tools that support active scanning and can help detect exposed ports include IP Scanner, Vulnerability Audit, Port Scanner, NS Lookup, and Trace Route

(Foong, Juremi, & Nathan, 2019). Passive scanning should be implemented to monitor regular network activity and check the software and patch versions on devices connected to the network.

Cryptographic algorithms should be used within the organization to secure data. These algorithms transform plaintext into ciphertext, so that data cannot be interpreted by unintended users or attackers. There are three major types of cryptographic algorithms: symmetric encryption, asymmetric encryption, and hash functions. Symmetric encryption algorithms, also known as secret-key algorithms, use one key for encryption and decryption. This means that if the key gets compromised, it can be used to decrypt anything encrypted with that key (Turner, 2019). These types of algorithms should be avoided when possible. Instead, the organization should use asymmetric algorithms, also known as public-key algorithms, whenever possible. Unlike symmetric encryption algorithms, these algorithms use different keys for encryption and decryption. This provides more protection than symmetric algorithms, but also requires more computing power (Wills, 2019). Hash functions will also be important for protecting the integrity of data. Hash functions do not use keys, but instead compute a fixed-length hash value based on the plaintext. This method is known as one-way encryption because it is impossible to recover the contents of the plaintext from the hash value (Kessler, 2020).

Examples of common encryption algorithms include the Caeser Cipher, AED, 3DES, RSA, and ECDSA (Lake, 2018). The Caeser Cipher involves shifting each letter a fixed number of spaces. AED, the Advanced Encryption Standard, is a complex symmetric encryption algorithm commonly used to encrypt communication data. The 3DES algorithm stands for triple DES, and is a more reinforced version of the original DES algorithm that runs the data through the algorithm three times. RSA stands for Rivest-Shamir-Adleman. This was the first widely accepted public-key algorithm. It is used in security protocols like PGP and TLS. Finally,

ECDSA stands for the Elliptical Curve Digital Signature Algorithm. This is a public-key algorithm that is a variation of the original DSA algorithm (Lake, 2018). There are other algorithms available as well, so care should be taken when selecting the one that is most appropriate for the organization and task.

Another way that the organization can increase security and reduce the probability of vulnerabilities is through the application of best practices during software development. According to *Rules versus Recommendations* (Java) (2015), code should follow guidelines, which are recommendations that increase code quality by enhancing the" safety, reliability, or security of a system." Noncompliant code is code that violates the coding guidelines, and therefore puts the system at risk. In place of noncompliant code, a compliant solution should be implemented, which follows the secure coding rules. Some of the secure coding rules discussed by Carnegie Mellon University (Flynn, 2015) include rules related to input validation and data sanitization, class declarations and initialization, expressions, numeric types and operations, characters and strings, object orientation, methods, and many others. An example of a secure coding rule is to normalize strings before validating them. For example, coders should forbid <script> tags in inputs in order to avoid cross-site scripting vulnerabilities (Mohindra, 2017). This is an example of a runtime vulnerability, which attackers can abuse. Many applications today use code injection at runtime to customize features and control parameters. This can be exploited through malicious code injections that allow a hacker's embedded logic to execute (Wills, 2019). Furthermore, transparency in the SDLC, utilization of an IDE, and configuration management tools can minimize vulnerabilities caused by compile and runtime errors.

Aside from utilizing secure coding practices, the security development lifecycle described by Microsoft (2020) should be implemented in order to reduce vulnerabilities that arise

from the software development process. The Microsoft Security Development Lifecycle (SDL) consists of twelve practices. The first step is to provide training so that everyone, including product managers, have an understanding of security basics. The second step is to define security requirements for the software project. The third step is to define metrics and compliance reporting. This involves agreeing on standards for identifying, communicating, and handling issues throughout the project. The fourth step is to perform threat modeling in order to identify potential vulnerabilities, determine the business impact of those threats, and then choose ways to mitigate them. The fifth step in the SDL is to establish design requirements. This process involves selecting security features such as cryptography, authentication, or logging. Step six in the cycle is to define and use cryptography standards. Step seven involves managing the security risk of using third-party components. Step eight is to use pre-approved development and testing tools. Step nine is to perform static analysis security testing. This step should be integrated into the commit process so that the source code is evaluated before it is compiled. Step ten is to perform dynamic analysis security testing. Dynamic testing is used to detect vulnerabilities at runtime, and can be accomplished through testing tools or suites. Step eleven is to perform penetration testing. This step involves simulated attacks carried out by security professionals to mimic the actions of hackers and reveal potential vulnerabilities. Finally, step twelve is to establish a standard incident response process (Microsoft, 2020).

Finally, all organizations should have plans for incident management and disaster recovery. The NIST (National Institute of Standards and Technology) incident response framework is a well-respected incident management approach that should be implemented within the organization. This framework includes the following steps: detection, response, mitigation, reporting, recovery, remediation, and lessons learned (Wills, 2019). The first step, detection,

involves analyzing and characterizing an event. Not all events are considered security incidents. Events can be deemed to have occurred due to natural causes, accidents, system failures, or intrusions. Intrusions typically lead to an incident management response. After an incident is detected, the response phase involves gathering data and information, asking questions, and determining who is in charge of the incident management process (Wills, 2019). Next, the incident is mitigated through containment and eradication to prevent further system damage or loss of data. The reporting phase involves keeping a running incident response log that communicates what happened, decisions that were made, and actions that were taken. During the recovery phase, systems are restored to their pre-attack state and management is notified that the systems are back up. Remediation involves taking steps to improve system security moving forwards. For example, software patches may be applied and configurations updated (bmc.com, 2020). Finally, the lessons learned phase involves documenting take-aways, information sharing, and coordination. Wills (2019) states that organizations should, “Review your incident response procedures for what worked and what didn’t, and update accordingly.”

The organization should also have a disaster recovery plan. An eight step disaster recovery plan is described by Schiff (2016): 1) Inventory hardware and software--it's important to document what you have so that you know what may have been compromised during an incident, and so that you can quickly get systems back up and running; 2) Define the tolerance for downtime and data loss--this section of the plan should define the recovery point objective (RPO) and recovery time objective (RTO), and what kinds of solutions you might need to achieve those constraints. Applications can also be prioritized in this part of the plan regarding which need to be restored first; 3) Define who is responsible for what and identify backup personnel--key roles and responsibilities show be defined here so there is a clear understanding

of who needs to do what. Individuals included in this plan should also have backups in case they are out of office; 4) Create a communication plan--this section should define how employees, vendors, suppliers, and customers will get the information they need. There should also be a prepared written statement from the company that can be published to alert, inform, and comfort customers in the case of an emergency; 5) Let employees know where to go in case of emergency and have a backup worksite:--security incidents have the potential to interrupt the normal flow of business. However, having a designated backup worksite will alleviate stress and planning during the time of an incident; 6) Make sure service level agreements (SLA's) include disasters/emergencies--it is important when outsourcing services or technology that the contract include acceptable timeframes for addressing incidents and restoring service; 7) Define how to handle sensitive information--this section should include procedures for maintaining and accessing sensitive information during an incident; and 8) Test the plan regularly--ensure that backup hardware and other components of the DR plan are working. Define how the plan will be tested and how often. Test employees to ensure they know their part in the DR plan (Schiff, 2016).

The best way for organizations to avoid security vulnerabilities is to anticipate them. Having clear security policies and procedures, as well as incident management and disaster recovery plans, will allow organizations to deter more threats and respond to incidents more quickly and efficiently. It is imperative that organizations support data confidentiality, integrity, availability, non-repudiation, and authentication (CIANA) in order to gain and maintain stakeholder trust. Furthermore, advocating for security awareness throughout the organization will help increase security and reduce risk. Some of the easiest components of security policies and procedures that organizations can implement include acceptable use policies, password

policies, security awareness policies, and authorization and authentication policies. In addition to these policies, thorough network management and monitoring, vulnerability scans and audits, and data encryption can help support CIANA. Furthermore, integrating the software development lifecycle with the security development lifecycle and implementing secure coding practices will reduce risk. Lastly, establishing clear incident management and recovery plans up front will allow organizations to more efficiently detect, handle, and recover from security incidents.

Resources

Bmc.com. (2020). What Is Threat Remediation? Threat Remediation Explained. Retrieved June 3, 2020 from https://www.bmc.com/blogs/what-is-threat-remediation-threat-remediation-explained/

Flynn, L. (2015). Rules versus Recommendations (Java). Carnegie Mellon University: Software Engineering Institute. Retrieved May 30, 2020 from https://wiki.sei.cmu.edu/confluence/pages/viewpage.action?pageId=88487386

Foong, L. L., Juremi, J., & Nathan, Y. (2019). Net Scan: Web-based Network Scanning Tool. International Journal of Psychosocial Rehabilitation, 23(4), 1238–1250. https://doi-org.proxy-library.ashford.edu/10.37200/ijpr/v23i4/pr190450

Lake, J. (2018). Encryption, hashing, salting – what's the difference? Comparitech. Retrieved June 3, 2020 from https://www.comparitech.com/blog/information-security/encryption-hashing-salting/

Microsoft. (2020). What are the Microsoft SDL practice? Retrieved May 29, 2020 from https://www.microsoft.com/en-us/securityengineering/sdl/practices#practice11

Mohindra, D. (2017). IDS01-J Normalize strings before validating them. Carnegie Mellon University: Software Engineering Institute. Retrieved May 30, 2020 from https://wiki.sei.cmu.edu/confluence/display/java/IDS01-J.+Normalize+strings+before+validating+them

Orrill, J. (2020). What is the Difference Between Active & Passive Vulnerability Scanners? Retrieved May 23, 2020 from https://smallbusiness.chron.com/difference-between-active-passive-vulnerability-scanners-34805.html

Schiff. J. (2016). 8 ingredients of an effective disaster recovery plan. Cio.com. Retrieved June 6, 2020 from https://www.cio.com/article/3090892/8-ingredients-of-an-effective-disaster-recovery-plan.html?page=2

Sveikauskas, D. (2018). Security by Design Principles according to OWASP. ThreatPress. Retrieved June 3, 2020 from https://blog.threatpress.com/security-design-principles-owasp/

Sargent, S. A., & Webb, J. P. (2020). The Key to Trust: Social Engineering Fraud and Modern Threat Detection. Benefits Magazine, 57(1), 22.

Turner, D. (2019). Summary of Cryptographic Algorithms- According to NIST. Retrieved May 20, 2020 from https://www.cryptomathic.com/news-events/blog/summary-of-cryptographic-algorithms-according-to-nist

Wills, M. (2019). (ISC)2 SSCP Systems security certified practitioner: Official study guide (2nd ed.). John Wiley & Sons.