Addressing challenges of security risk management in computer security

The practice of security risk management (SRM) begins with a thorough and well-thought-out risk assessment [1]. In order to identify risks, we need to consider the assets that are important to the organisation and / or individual, what threats they may face, and what vulnerabilities exist that could be exploited [2]. For an organisation (or individual), identifying the *most important* asset should not be too difficult, as it is generally related to what they specialize in: banks – money, social media sites – data, manufacturers – intellectual property or the products in their warehouses perhaps.

However, the challenge comes when we attempt to identify and evaluate all other assets, such as customer data, machinery, hard-copies of information, hardware, employee emails, intangibles (software, intellectual property, reputation). It is a much more difficult task to correctly evaluate these other assets, as they have different priorities to different people within the organisation. For example, stakeholders may place more emphasis on the reputation of the organisation, rather than the security of their email system. Additionally, placing a value on intangible assets can be more challenging as more factors have to be considered, such as recovery costs, replacement costs and indirect costs, which can be easily overlooked.

Failing to correctly evaluate, or identify an asset in the first place, can lead to issues in the future. Some threats can be less common / likely in the present than they may be in the future. An asset that is given low priority today because the chance of an attack on it is very low, can result to a potential exploit in the future, once attackers find a way to use that asset. If no control measures have been taken in the past, the attacker will have a very easy time.

While it is very difficult to take all risks into account and create solutions to prevent or mitigate them, a way to minimize their impact is to monitor and manage the risks as time goes on, while keeping in mind the " fast-changing environment, which renders risk assessment and management in the information technology sector so difficult" [3]. Reviewing and reassessing risks is important in order to ensure that they are assessed to current requirements, standards and issues present to the organisation.

This is a challenging task, and there may be cases where risk managers fail to correctly monitor or re-assess risks. In such cases, a system monitor can help detect attacks on assets that have not been correctly re-assessed. System monitoring provides a capability that allows detection of actual or attempted attacks, and is essential in order to effectively respond to attacks [4], which can enable you to detect and react to attacks.

- Peterson, K. (2010), *The professional protection officer* . Oxford: Butterworth-Heinemann, p. 316.
- Nurse, J. (2019), *Lecture Slides CO634-3-Risk-Management. pdf* . s. 13.
- Stefan Fenz Johannes Heurix Thomas Neubauer Fabian Pechstein, (2014), *Current challenges in information security risk management* , Information Management & Computer Security, Vol. 22 Iss 5 pp. 410 – 430
- *10 steps to cyber security* , https://www. ncsc. gov. uk/collection/10-steps-to-cyber-security? curPage=/collection/10-steps-to-cyber-security/the-10-steps/monitoring, Accessed: Nov 2019

What makes a good authentication scheme and why? What is your preferred scheme and why is that better than other schemes?

A good authentication scheme is a scheme that is appropriate for the resources it protects. In order to decide whether an authentication scheme is good for a particular system, we need to consider what kind of information is stored in the system. For example, your home computer may only be protected by a password, and you may use two-factor authentication to log in to your email. Meanwhile an organisation that values its resources may have pin-pads as well as RFID cards and some form of biometrics.

There are three types of authentication [5]:

- Something I know
- Something I possess
- Something I am

A good authentication scheme would have at least a combination of two of the types mentioned above. For example, a password with two factor authentication combines ‘ something I know’ and ‘ something I possess’. The individual knows a password, and has a device or object that serves as the second method of authentication alongside the password (code sent via sms / email / an app, a smart card etc.).

I think this should be the minimum to define a ‘ good’ authentication scheme because it has less vulnerabilities. For example, if authentication only required an RFID card, the RFID card can easily be stolen. If it only required a password, the password can be guessed, brute-forced or social-engineered. But as long as there are at least two authentication requirements, the likelihood that the person attempting to access the resources is who they say they are, is significantly higher.

My preferred authentication scheme is a password with two-factor authentication, as it’s convenient and (in my opinion) secure enough for my needs. I use a password manager (KeePass) with a 32-character master-password that (supposedly, according to KeePass quality estimation) has 210 bits of entropy. This allows me to easily generate strong unique passwords for all of my online accounts, whether they implement two-factor authentication or not. It prevents me from reusing passwords between accounts, which minimises the impact when one website has a data-breach and my password gets stolen. While there are risks like my cloud storage (on which I store the database) being breached, I’m not too worried about anyone reading the contents.

On the other hand, as previously mentioned, I don’t believe an authentication scheme can be ‘ better’ than others in a general sense, since we need to consider what we need to protect. A password may be enough for the average user, while the CEO of a company may use several layers of authentication to access their laptop or authenticate into their company’s systems.

[5] Nurse, J. (2019), *Lecture Slides CO634-5-Symmetric-Authentication-Intro. pdf* . s. 6.

Describe known attacks against the Diffie-Hellman protocol, and the most common countermeasures to stop them. Consider the date of the attacks and changes over time that reduced or increased their relevance.

The Diffie-Hellman protocol is a method for two parties (A and B) to generate a shared key that they can use to communicate over an unsecure channel.

Man-in-the-middleattack is a popular attack method on the Diffie-Hellman protocol. When A and B attempt to establish a communication, a third party (C) interferes and impersonates both parties. The third party will impersonate B while talking to A, and it will impersonate A while talking to B. The two parties are not aware this is happening because Diffie-Hellman does not provide a method of authentication to prove your identity when you start a conversation.

One of the most common countermeasures against a man-in-the-middle attack is to implement some form of authentication. This can be used to ensure that the message comes from a legitimate source. A way of implementing this is by using public key cryptography, where each party has a private and public key. This allows A to encrypt their identity with the public key of B, after which B can decrypt it using their own private key.

A popular man-in-the-middle attack is the Logjam vulnerability / attack, that allows an attacker to downgrade vulnerable Transport Layer Security connections to 512-bit export-grade cryptography, that allows them to read and modify any data passing over the connection [6].

Denial-of-Serviceattacks are also potentially damaging attacks that consists of flooding one party with bogus requests with forged IP addresses. This can be done either by targeting the initiator or the responder in a DH key negotiation. The constant requests to the target to instigate a new connection or random numbers to be used for a shared key. This causes the target to waste computational power and possibly system failure.
One way of dealing with this type of attack is to use stateless cookiesthat

allows the party receiving a connection request to verify the legitimacy of the initiator [7]. This is done by the receiving party sending some data (the stateless cookie) and asking the initiator to retransmit the request while including the cookie.

Lastly, replayattacks are a form of attack where the attacker takes a previously sent message and sends it again in order to attempt to replicate the interaction [8]. If the receiving party is not aware / does not check for this kind of attack, it can be very effective as there would be no suspicion. For example, in a conversation between A and B, B requests A to prove her identity. C can intercept the identity message from A to B and save it. Even though C cannot read the message in any way, it can be used in the future to initiate a conversation between C and B, because C has the (possibly hashed) identity of A.

To defend against replay attacks, session IDs or cryptographic nonces can be used, which are unique per session. The nonce is a random number encrypted by the receiver and sent to the initiator that does a simple operation on it and sends it back. This helps to prove the fact that this new communication being established is new and it is not a replay attack.

- Weak Diffie-Hellman and the Logjam Attack, https://weakdh. org/, Accessed: Nov 2019
- Fathirad, I., Devlin, J. and Atshani, S. (2016). *Network-Specific Attacks on Diffie-Hellman Key-Exchange in Commercial Protocols* .
- Raymond, J. andStiglic, A. (n. d.). *Security Issues in the Diffie-Hellman Key Agreement Protocol* .

What is BlueKeep (CVE-2019-0708) and why is it considered as such a serious vulnerability?

BlueKeep is a remote code execution vulnerability discovered in Microsoft's Remote Desktop Protocol implementation. In order to exploit this vulnerability, an attacker needs to connect to the target system using RDP and send specially crafted requests [9].

- *CVE-2019-0708 | Remote Desktop Services Remote Code Execution Vulnerability* (2019), https://portal. msrc. microsoft. com/en-US/security-guidance/advisory/CVE-2019-0708

How can attackers bypass firewalls? Describe at least 3 possibilities providing enough technical details and some tools and countermeasures, if applicable.

The first method I will be talking about ispiggybacking. An attacker can penetrate the security system of a trusted third party, and either steal data from them that can be used in future attacks, such as targeting employees to gain their credentials. For example, attack on health insurance provider Anthem Inc that happened in 2014 started from a single user within one of Anthem's subsidiaries opened a phishing email containing malicious content [refhere], that allowed the attackers to gain remote access to several computers within the organisation.

Alternatively, organisations allow their employees to work remotely while not on company property by providing a VPN service that they can use to connect to the company's systems and access their work. An attacker can penetrate the home-user's system or, even easier, break into their home and steal their device, after which they may be able to use that VPN to access the corporate network. Additionally, the employee may be storing sensitive data directly on the device that can be used maliciously. Countermeasures for such an attack can include restricting VPN access to critical systems only, or defining a policy that classifies users based on their needs and the type of data they handle (how confidential it is) [anotherref]. Supplying separate work-machines that employees can use to connect with a VPN, rather than using their personal computers can be an additional measure to ensure (and force) employees to use their work computer for work, and avoid keeping personal files on that machine.

The second method I will describe isDNS Tunnelling. DNS Tunnelling uses the DNS protocol to encode the data of other programs or protocols in DNS queries and responses over port 53. [refmore]. Attackers know that enterprise network defences allow DNS traffic over port 53, so they manipulate DNS requests to exfiltrate data from a compromised system to their own infrastructure [something].

A popular tool is Iodine, and it lets you tunnel IPv4 data through a DNS server. An attacker acquires a domain and configures the domain name servers to his own DNS server such that it points to his machine, and they are able to establish a bi-directional data transfer channel. The hacker easily contaminates the computer because the DNS requests are always allowed to move in and out of the firewall [referencenew].

To combat this, the basic idea is to carefully monitor all traffic coming in and out of the network, and attempting to identify data extraction techniques and take appropriate actions. There are several tools that can help with this, such as TunnelGuard, ZScaler, Splunk and many more that provide useful protection functionality such as content filtering, advertisement blocking, malware and phishing

Lastly, attackers can exploitvulnerabilitiesin software, which can enable them to bypass firewalls and access data. Vulnerabilities can arise either from intended program features or from program errors. For example, Microsoft’s Double Kill vulnerability is a remote code execution flaw residing in Windows VBScript which could be exploited through Internet Explorer. It allowed an attacker set up multiple backdoors on target machines that enabled them to receive more commands after the initial intrusion, and they were able to access and overwrite the whole user space memory address [refhere]. The countermeasure to such vulnerabilities is to install patches as soon as possible, if available.

- https://www. cableforum. uk/board/showthread. php? t= 28148
- https://www. insightsforprofessionals. com/it/security/hacks-sure-to-defeat-your-firewall
- https://www. mrg-effitas. com/research/bypass-hardware-firewalls-def-con-22/
- https://www. linuxschoolonline. com/ssh-reverse-port-forwarding-or-how-firewalls-can-be-bypassed/
- [refhere] McGee, M. (2017), *A new In-Depth Analysis of Anthem Breach,* https://www. bankinfosecurity. com/new-in-depth-analysis-anthem-breach-a-9627, Accessed: Nov 2019

- [refmore] *What is DNS Tunnelling* , https://www. infoblox. com/glossary/dns-tunneling/, Accessed: Nov 2019
- [anotherref] McCormick, J. (2002), *Learn how VPN users can open a hole in your network,* https://www. techrepublic. com/article/lock-it-down-dont-let-vpn-users-be-the-weak-link-in-network-security/, Accessed: Nov 2019
- [something] *DNS Tunnelling – How to stop attackers from using port 53 for data exfiltration and command & control callbacks* , https://learn-umbrella. cisco. com/solution-briefs/dns-tunneling, Accessed: Nov 2019
- [iodine] *Iodine* , https://github. com/yarrick/iodine
- [referencenew] Taylor, K., *How to Prevent DNS Tunnelling* , https://www. hitechnectar. com/blogs/prevent-dns-tunneling/, Accessed: Nov 2019
- newref] Yin, D. (2018), *An Analysis of the DLL Address Leaking Trick used by the " Double Kill" Internet Explorer Zero-Day exploit (CVE-2018-8174),* https://www. fortinet. com/blog/threat-research/analysis-of-dll-address-leaking-trick-used-by-double-kill-internet-explorer-0-day-exploit. html, Accessed: Nov 2019