PROJECT 1: Windows 10 Vulnerability & Patch

Andrew Pickner
Luke Joyce

2/6/20

1. (5 points) In 2-3 sentences, explain what your event was about at a conceptual level. In this, explain how it relates to cybersecurity.

Microsoft's operating system Windows 10 (as well as Windows Server 2016/2019) was reported to have a 'critical vulnerability' by the National Security Agency. The NSA made the swift decision to alert Microsoft and the public about the bug because it could've left over 900 million personal computers vulnerable to attack. The vulnerability presented a large risk to anyone using the Windows software because it allowed an adversary to compromise a system in a few ways.

2. (5 points) In 2-3 sentences, explain some of the technical details of your event.

The vulnerability abuses Microsoft's encryption API for Elliptical Curve Cryptography and undermines how Windows verifies cryptographic trust. More specifically, the vulnerability allows an adversary to add a verified signature to malicious code or allows Man-in-the-Middle attackers to decrypt confidential files. Not only could a skilled threat actor figure out how to exploit the vulnerability rather quickly, but the fact that Windows 10 is the most widely used operating system in the world makes this vulnerability so severe by NSA standards.

3. (5 points) What is the most interesting thing you took away from reading about your article?

Typically the NSA withholds what they call 'critical vulnerabilities' so they can conduct further research or weaponize the vulnerability themselves. In this case, whether it be a change in agency policy or this vulnerability truly is *that* critical, I found it interesting that the NSA chose to disclose the information so quickly.

4. **(5 points) Analyze some impacts from your event on each of the CIA triad in 1-2 sentences each.**

    - **Confidentiality:** Due to the ability to exploit code signatures, a threat actor could put ransomware on a system with much more ease which could lead to someone's data being encrypted and later disclosed that data. Man-in-the-middle attacks expose someone to a similar situation where their encrypted communications are decrypted and later disclosed.

    - **Integrity:** Again, exploiting code signatures could allow a threat actor to easily put a malicious program on a user's system that corrupts data. The man-in-the-middle aspect of this vulnerability allows the threat actor to decrypt communications which would allow for a much easier process for changing the integrity of a message.

    - **Availability:** If a threat actor has an easier way to put ransomware on a system, then this could greatly impact the availability because the user may lose access to all of their data unless the ransom is paid. The Man-in-the-middle aspect could allow the threat actor in the middle to stall or cease communications entirely which *could* be argued to impact integrity, but we argue that ceasing communications entirely would be affecting the availability more.

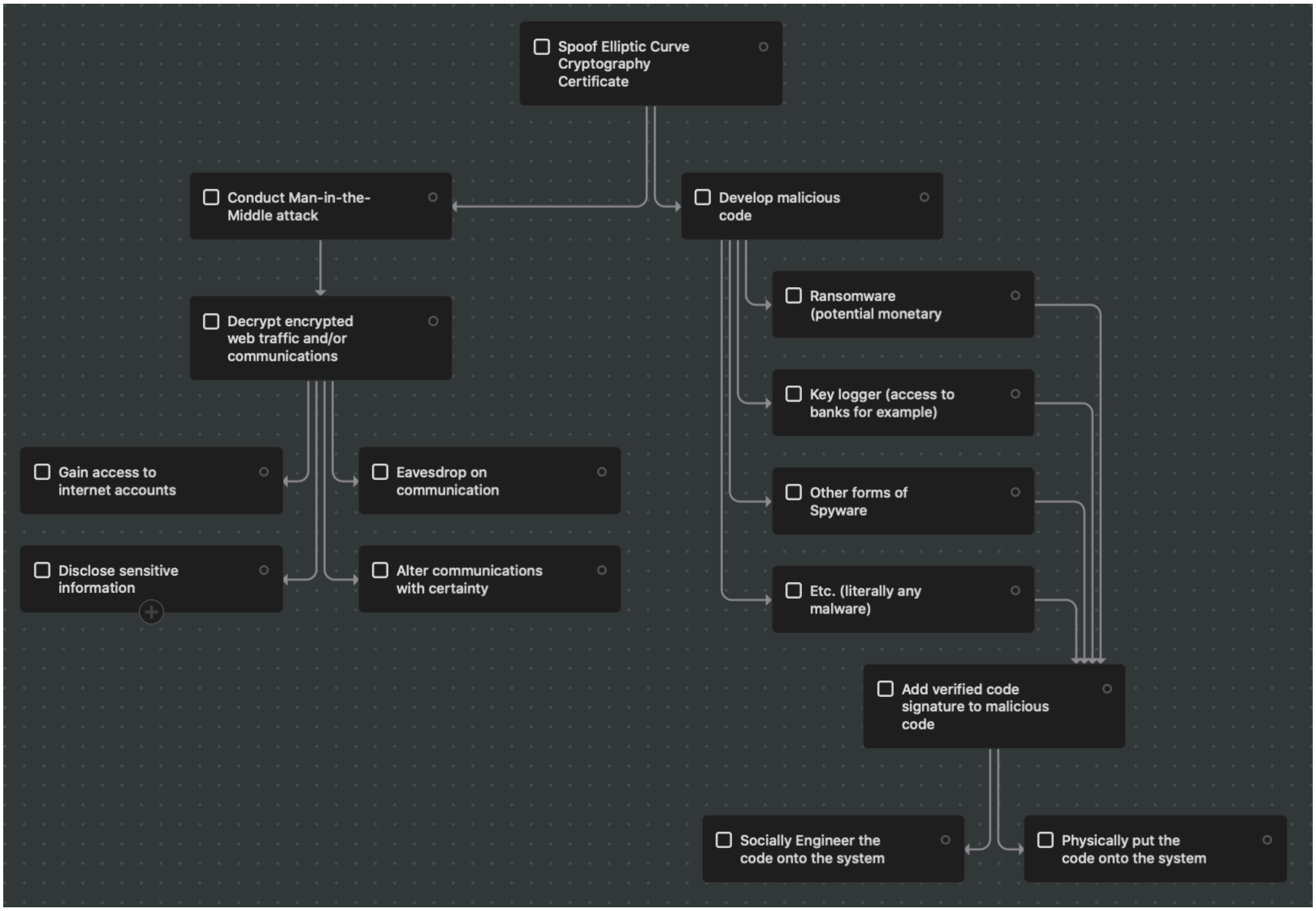
5. **Pick 3 common design principles and explain the impact on your event in 1-2 sentences each.**

    - **Open-Design:** This design principle applies directly to this situation because the vulnerability deals with Microsoft's CrypotAPI which is their proprietary API software that integrates into their operating system. With Microsoft's CryptoAPI an open design may not be possible, but as open design as possible should be the goal.

    - **Defense-in-Depth:** Although this is a tricky thing to protect against, it seems that Microsoft *could* have added an additional layer of security to their cryptographic API or the surrounding functionality. However, this design principle can impact the usability of the software which is one potential reason that Microsoft didn't go about fixing the vulnerability using this principle.

- **K.I.S.S.:** Lastly, I'm not 100% on this one, but based on the previous design principles discussed, this may be the most crucial one of all. If Microsoft can't make their CryptoAPI more open, or they can't add depth to their defenses, they should strive to keep their API system as simple as possible minimizing human programming errors.

6. **(3 points) Create an attack tree of your current event (15-20 leaf nodes is fine). You may create it digitally or write it neatly and scan it in - whichever is easier for you.**



Spoof Elliptic Curve Cryptography Certificate
Conduct Man-in-the-Middle attack
Develop malicious code
Decrypt encrypted web traffic and/or communications
Ransomware (potential monetary
Key logger (access to banks for example)
Gain access to internet accounts
Eavesdrop on communication
Other forms of Spyware
Disclose sensitive information
Alter communications with certainty
Etc. (literally any malware)
Add verified code signature to malicious code
Socially Engineer the code onto the system
Physically put the code onto the system

Citations

1. **https://www.wired.com/story/nsa-windows-10-vulnerability-disclosure/**

    - The first article examined to gain a basic understanding of the problem. Also provides external sources 2 and 3.

2. **https://media.defense.gov/2020/Jan/14/2002234275/-1/-1/0/CSA-WINDOWS-10-CRYPT-LIB-20190114.PDF**

    - A brief explanation of the vulnerability and why the NSA believed it to be so critical. Very helpful in understanding some of the technical aspects of the vulnerability as well.

3. **https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/CVE-2020-0601**

    - A brief explanation of the vulnerability from the viewpoint of Microsoft. Sorta helpful in understanding some of the technical aspects of the vulnerability as well.

4. **https://www.washingtonpost.com/national-security/nsa-found-a-dangerous-microsoft-software-flaw-and-alerted-the-firm--rather-than-weaponize-it/2020/01/14/f024c926-3679-11ea-bb7b-265f4554af6d_story.html**

    - Another non-technical explanation of the problem.

5. **https://www.washingtonpost.com/news/powerpost/paloma/the-cybersecurity-202/2020/02/06/the-cybersecurity-202-here-s-why-nsa-rushed-to-expose-a-dangerous-computer-bug/5e3b0f41602ff15f8279a52e/**

    - More insight as to why the NSA felt so strongly about releasing the information about the vulnerability.

I, Andrew was the typist while Luke sat by and helped me reason through the answers.