THE SOLARWINDS ATTACK

Abstract: One of the most sensational cyberattacks, the SolarWinds Attack was aimed at breaching America's confidential information. It is suspected to be a state-sponsored attack since the group accused of the attack, CozyBear, is a hacker group believed to be associated with Russian intelligence agencies. The SolarWinds attack is one of the most successful ones considering the impact and duration of the attack. This paper aims to analyze the SolarWinds attack from the Cryptographic and Network Security point of view. The techniques used by the hackers to execute the attack and the remedial measures taken up by the affected parties will be elaborated.

Keywords: ***Cyberattack, Supply Chain Attack, Cyber espionage, Compromised updates, Remote Access Tool, KillSwitch, Cyber Audit***

The SolarWinds Attack

1. Introduction:

The SolarWinds attack was one of the worst cyber-attacks that America had ever suffered. Believed to be state sponsored, it was one of the most sophisticated and well planned. Investigations are still continuing to reveal intriguing aspects this attack. The Departments of Defense, Energy(which also includes America's nuclear arsenal), Homeland Security, several Fortune 500 companies are only few of the high profile victims of this attack. The discovery of the attack happened in one of the worst possible time; when America was polarized by the Presidential Election and the world was afflicted by the Corona virus. The report here aims to perform a comprehensive analysis of the SolarWinds attack. Firstly, the list of related works has been stated. Next, a deep dive on the attack will be performed: the cause, type and loss inflicted will be elaborated. Finally, we discuss the security measures taken up and also discuss the suggestible practices/measures that are required to prevent such attacks in the future.

2. Related Work:

1. Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing: https://www.researchgate.net/publication/357541175_SolarWinds_Software_Supply_Chain_Security_Better_Protection_with_Enforced_Policies_and_Technologies

2. IEEE 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT): https://ieeexplore.ieee.org/document/9579611

3. E-International Relations research paper: https://www.e-ir.info/2021/06/17/the-SolarWinds-attack-and-its-lessons/

4. International Journal of Safety and Security Engineering: https://iieta.org/journals/ijsse/paper/10.18280/ijsse.110505

5. TechTarget.com article: https://www.techtarget.com/whatis/feature/SolarWinds-hack-explained-Everything-you-need-to-know

6. Deep dive into Solorigate by Microsoft: https://www.microsoft.com/en-us/security/blog/2021/01/20/deep-dive-into-the-solorigate-second-stage-activation-from-sunburst-to-teardrop-and-raindrop/

7. CEO of SolarWinds blogpost: https://orangematter.SolarWinds.com/2021/01/11/new-findings-from-our-investigation-of-sunburst/

8. Sunburst Attack Flow, C2 protocol & Prevention: https://www.cynet.com/attack-techniques-hands-on/sunburst-backdoor-c2-

communication-protocol/

9. Senate Testimony Crowdstrike: https://www.crowdstrike.com/wp-content/uploads/2021/03/george-kurtz-senate-testimony-on-cybersecurity-and-supply-chain-threats-022321.pdf

10. GOLDEN SAML TECHNIQUE EXPLAINED: https://owasp.org/www-chapter-singapore/assets/presos/Deconstructing_the_SolarWinds_Supply_Chain_Attack_and_Deterring_it_Honing_in_on_the_Golden_SAML_Attack_Technique.pdf

3. The Attack:

Chapter 1: The causes of the attack:

Background: SolarWinds Corporation is an American company that develops software for businesses to help manage their networks, systems, and information technology infrastructure. One of their products is Orion, which is an infrastructure monitoring and management platform designed to simplify IT administration. All IT equipment such as servers, routers, switches and workstations are monitored on a single authorized machine. The platform detects and displays any problems to the admin and also provides remedial measures. In 2020, according to the company's homepage there were about 300,000 customers which also included high profile ones such as:
(as stated in their homepage)

- More than 425 of the US Fortune 500
- All ten of the top ten US telecommunication companies
- All five branches of the US Military
- The US Pentagon, State Department, NASA, NSA, Postal Service, NOAA, Department of Justice, and the Office of the President of the United States

  …..

The SolarWinds Orion platform is a third-party tool in the cyber systems of the customers.

Initial Remarks: SolarWinds must not have publicly mentioned the Government of United States as one of its customers. This is because it generates interest in hostile nation states to find weaknesses in the system, hack it and then exploit the system for malicious purposes.

Timeline: The timeline for the attack as stated by the SolarWinds company and also confirmed by Microsoft is as given below:

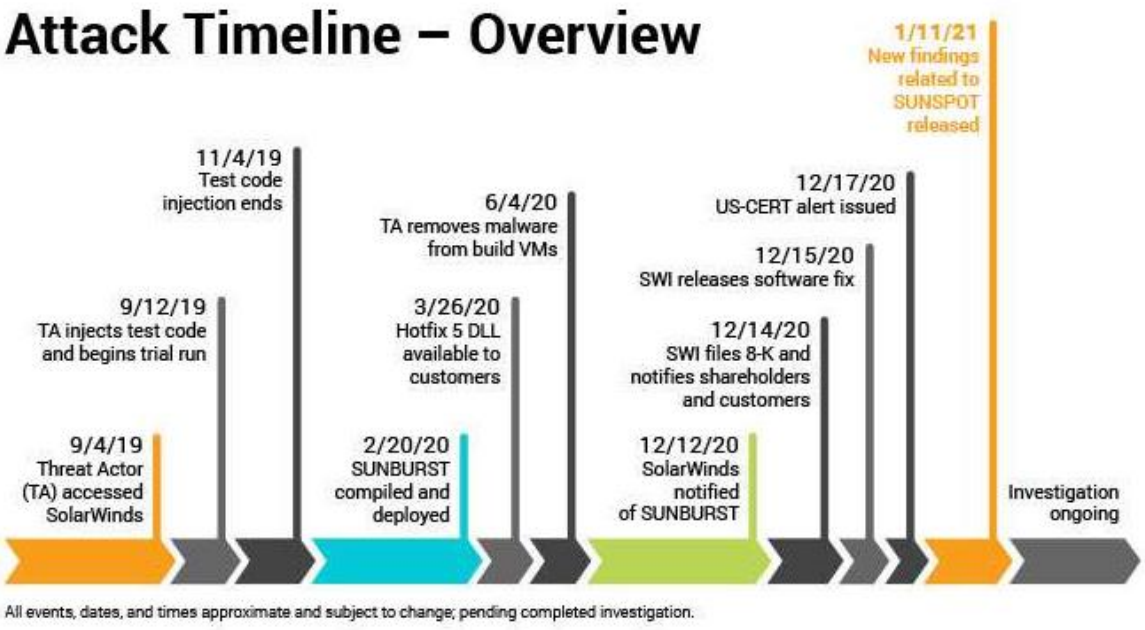

Fig. 1: The attack timeline as released by SolarWinds

However, according to an interview with Sudhakar Ramakrishna, the CEO of SolarWinds, hackers may have gained access to SolarWinds as early as January. This fact is especially startling because it means that the hackers would have two years' worth of access to the systems of SolarWinds and their customers.

Perpetrators: The attack is said to have been carried out by CozyBear a Russian hacking group believed to be associated with one or more intelligence agencies of Russia, especially the Foreign Intelligence Service (SVR). The attacker was known by different names: UNC2452(Uncategorized groups) and APT29(Advanced Persistent Threat). As per Mandiant.com, both these names have now been merged.

Objective: The attack is said to be an act of Cyber-Espionage by Russia, according to Law Professor Jack Goldsmith. However, Microsoft president Brad Smith termed it as a cyber-attack. In both these types, the common goal is to weaken the United States of America and related parties.

Chapter 2: Type of the Attack:

This is the most elaborate chapter of this report. Here we thoroughly analyze the mechanism and techniques used by the hackers to execute the attack.

The attack comes under the category of supply chain attack. A supply chain can be depicted as follows:

Fig. 2: A supply chain model

Similar to a physical supply chain, we have the software supply chain which can be depicted as follows:

Fig. 3: A software supply chain model

In a supply chain attack, the attackers focus on the weakest/ most vulnerable part of the supply chain for a software and hence try to compromise all the remaining phases of the supply chain that is, distribution and feedback. In this case, the sourcing part where the their party vendor SolarWinds provided Orion as a component proved to be the attacking point for the hackers. All they had to do was to compromise the Orion software in such a way that when Orion is embedded in the internal systems of the customers, such systems are also compromised. Hence Orion was to become a gateway to the customers' systems.

According to the timeline, the access to the Orion software began in September 2019. However, during the RSA conference 2021, Sudhakar Ramakrishna, the CEO of SolarWinds, said that his forensic investigation team had evidence to suggest that the access may have happened as early as January 2019. Since the attack was discovered only in December 2020, it meant that the hackers had two years of access to the SolarWinds systems and potentially its customers too.

There were four malwares that were responsible for the compromise of the cyber systems of SolarWinds and its customers. They are as stated below:

1. *sunspot*: CrowdStrike describes SUNSPOT as "a malicious tool that was deployed into the build environment to inject [the SUNBURST] backdoor into the SolarWinds Orion platform.". For the Orion software to work, there needs to be a compilation process. Before compilation, the code is audited by the *msbuild.exe* file to ensure that the code has not been tampered with. SUNSPOT monitors running processes for instances of MsBuild.exe. When SUNSPOT finds an MsBuild.exe process, it will spawn a new thread to determine if the Orion software is being built and, if so, hijack the build operation to inject SUNBURST. The monitoring loop executes every second, allowing SUNSPOT to modify the target source code before it has been read by the compiler.

A backdoor attack is a way to access a computer system or encrypted data that bypasses the system's customary security mechanisms. Hence the attackers established a mechanism that would enable them to gain access to the system whenever required.

To gain access to the customers' systems, the attackers used this malware to compromise the updates for Orion software. When the customers downloaded and installed the updated, their computers would be infected.

2. *sunburst*: This was the backdoor malware that was inserted into the customers' systems by compromising the updates.

The attackers inserted malicious code into SolarWinds.Orion.Core.BusinessLayer.dll, a code library belonging to the SolarWinds Orion Platform. The attackers had to find a suitable place in this DLL component to insert their code. Ideally, they would choose a place in a method that gets invoked periodically, ensuring both execution and persistence, so that the malicious code is guaranteed to be always up and running. Such a suitable location turns out to be a method named RefreshInternal.

```
internal void RefreshInternal()
{
    if (log.get_IsDebugEnabled())
    {
        log.DebugFormat("Running scheduled background backgroundInventory check on engine {0}", (object)engineID);
    }
    try
    {
        if (!OrionImprovementBusinessLayer.IsAlive)
        {
            Thread thread = new Thread(OrionImprovementBusinessLayer.Initialize);
            thread.IsBackground = true;
            thread.Start();
        }
    }
    catch (Exception)
    {
    }
    if (backgroundInventory.IsRunning)
    {
        log.Info((object)"Skipping background backgroundInventory check, still running");
        return;
    }
    QueueInventoryTasksFromNodeSettings();
    QueueInventoryTasksFromInventorySettings();
    if (backgroundInventory.QueueSize > 0)
    {
        backgroundInventory.Start();
    }
}
```

Fig. 4: The method infected with the bootstrapper for the backdoor

The modification to this function is very lightweight and could be easily overlooked—all it does is to execute the method OrionImprovementBusinessLayer.Initialize within a parallel thread, so that the normal execution flow of RefreshInternal is not altered.

Why was this method chosen rather than other ones? A quick look at the architecture of

this DLL shows that RefreshInternal is part of the class SolarWinds.Orion. Core.BusinessLayer.BackgroundInventory.InventoryManager and is invoked by a sequence of methods that can be traced back to the CoreBusinessLayerPlugin class. The purpose of this class, which initiates its execution with a method named Start (likely at an early stage when the DLL is loaded), is to initialize various other components and schedule the execution of several tasks. Among those tasks is Background Inventory, which ultimately starts the malicious code.

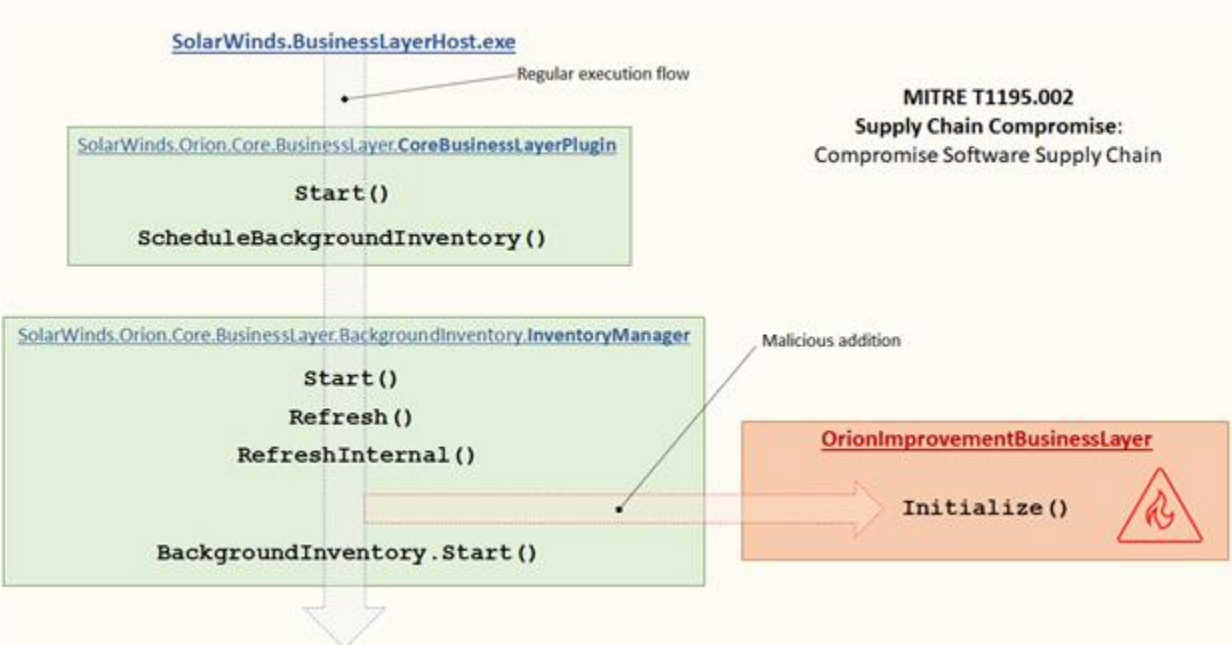

Fig. 5: The method infected with the bootstrapper for the backdoor

The functionality of the backdoor resides entirely in the class OrionImprovementBusinessLayer, comprising 13 subclasses and 16 methods. Its name blends in with the rest of the legitimate code. The threat actors were savvy enough to avoid give-away terminology like “backdoor”, “keylogger”, etc., and instead opted for a more neutral jargon. At first glance, the code in this DLL looks normal and doesn’t raise suspicions, which could be part of the reason why the insertion of malicious code was undetected for months, especially if the code for this DLL was not frequently updated.

To have some minimal form of obfuscation from prying eyes, the strings in the backdoor are compressed and encoded in Base64, or their hashes are used instead.

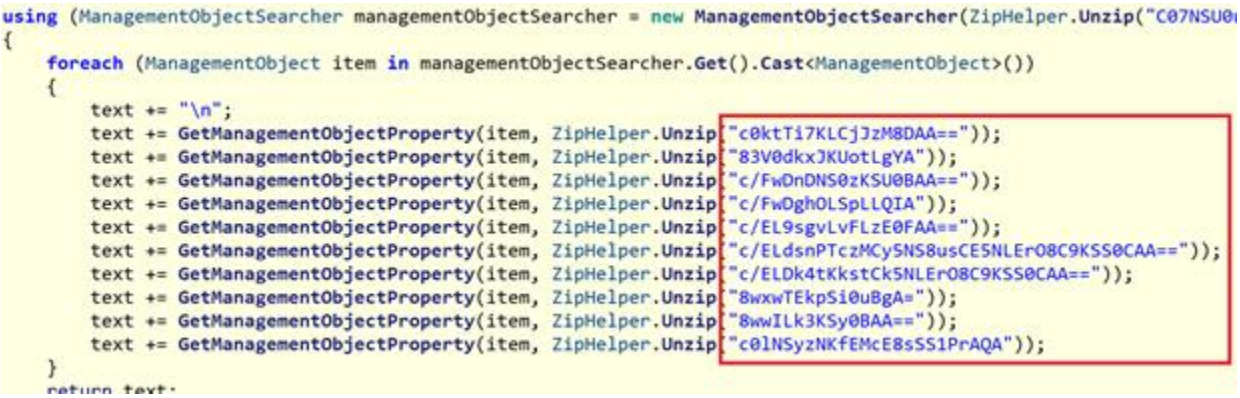

Fig. 6: Example of obfuscated strings

The Initialize method is the de facto execution entry point of the backdoor. It carries out several checks to verify that it is running in a real victim's environment:

- Verifies that the process hosting the malicious DLL is named SolarWinds.businesslayerhost.exe.
- Checks that the last write-time of the malicious DLL is at least 12 to 14 days earlier. This is because a software is usually monitored after their install to prevent technical issues arising out of the installation. Any malicious activity at this time would expose the backdoor malware.
- Delays execution by random amounts of time
- Verifies that the domain name of the current device meets the following conditions:

1. The domain must not contain certain strings; the check for these strings is implemented via hashes, so at this time the domain names that are block-listed are unknown
2. The domain must not contain "SolarWinds"
3. The domain must not match the regular expression (?i)([^a-z]|^)(test)([^a-z]|$), or in simpler terms, it must not look like a test domain

- Checks that there are no running processes related to security-related software (e.g., Windbg, Autoruns, Wireshark)

- Checks that there are no drivers loaded from security-related software (e.g., groundling32.sys)
- Checks that the status of certain services belonging to security-related software meets certain conditions (e.g., windefend, sense, cavp)
- Checks that the host “api.SolarWinds.com” resolves to an expected IP address

If any of these checks fail, the backdoor terminates. All these inspections are carried out to avoid exposing the malicious functionality to unwanted environments, such as test networks or machines belonging to SolarWinds.

After the extensive validation described above, the backdoor enters its main execution stage. At its core, the backdoor is a very standard one that receives instructions from the C2(Command and Control) server, executes those instructions, and sends back information. The type of commands that can be executed range from manipulating of registry keys, to creating processes, and deleting files, etc., effectively providing the attackers with full access to the device, especially since it’s executing from a trusted, signed binary.

In its first step, the backdoor initiates a connection to a predefined C2 server to report some basic information about the compromised system and receive the first commands. The C2 domain is composed of four different parts: three come from strings that are hardcoded in the backdoor, and one component is generated dynamically based on some unique information extracted from the device. This means that every affected device generates a different subdomain to contact (and possibly more than one).

The dynamically generated portion of the domain is the interesting part. It is computed by hashing the following data:

- The physical address of the network interface
- The domain name of the device
- The content of the MachineGuid registry value from the key HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography

The backdoor also generates a pseudo-random URI that is requested on the C2 domain. Like the domain, the URI is composed using a set of hardcoded keywords and paths, which are chosen partly at random and partly based on the type of HTTP request that is being sent

out.

Finally, the backdoor composes a JSON document into which it adds the unique user ID described earlier, a session ID, and a set of other non-relevant data fields. It then sends this JSON document to the C2 server. If the communication is successful, the C2 responds with an encoded, compressed buffer of data containing commands for the backdoor to execute. The C2 might also respond with information about an additional C2 address to report to. These commands allow the attackers to run, stop, and enumerate processes; read, write, and enumerate files and registry keys; collect and upload information about the device; and restart the device, wait, or exit.

Once backdoor access is obtained, the attackers follow the standard playbook of privilege escalation exploration, credential theft, and lateral movement hunting for high-value accounts and assets.

Since FireEye, a cybersecurity company, was also a user of Orion, hackers also gained access to the cyber systems of FireEye and stole Cobalt Strike, a penetration testing tool.

3. *teardrop*: CISA reports that TEARDROP is a malicious 64-bit dynamic-link library (DLL) that decrypts and loads a malicious payload from an embedded code buffer. When executed, the malware attempts to read the first 64-bytes of a file named festive_computer.jpg but it does not actually utilize the data it reads from this file and will continue executing even if the file doesn't exist. After attempting to read festive_computer.jpg, the TEARDROP DLL uses an XOR cipher to decrypt and execute the Cobalt Strike Beacon Implant (Version 4) remote access tool (RAT) contained within its embedded code buffer. TEARDROP does not create any files during this process since the malware operates entirely within memory.

Cobalt Strike is a legitimate penetration testing tool that has become increasingly popular amongst threat actors due to its wide array of powerful features. Its capabilities include keylogging, taking screenshots, deploying additional payloads, exploiting system vulnerabilities to facilitate additional attacks, evading detection with various countermeasures, rapidly exfiltrating data through encrypted tunnels, and more.

4. *raindrop*: While Teardrop was used on computers that had been infected by the original Sunburst Trojan, Raindrop appeared elsewhere on the network, being used by the attackers

to move laterally and deploy payloads on other computers. Although Raindrop is very similar to Teardrop, there are some key differences between the tools. As mentioned previously, Raindrop uses a different packer. It uses steganography to find and extract the payload. It makes use of AES and LMZA algorithms to decrypt and decompress the payload.

The main purpose of using raindrop and teardrop was to deploy Cobalt Strike and make full use of its features which makes it even easier to steal data, avoid detection and monitor weaknesses in networks of the customers.

The malware variants 2,3 and 4 together execute an attack known as the Golden SAML attack. In SAML (Security Assertion Markup Language), the basic construct is when a client attempts to authenticate with a service provider, they are redirected to an authentication server. Once authenticated, they are provided a cryptographically signed response that the client then provides back to the service provider. Once received, the response is validated.

Golden SAML is a technique that allows an adversary to generate their own SAML response with the content and authorizations they deem necessary. In the SolarWinds attack, since they had privileged access to the entire system and hence the certificates too. They can impersonate just about any user and privilege in the organization. Not only that, but they can do it from anywhere in the world. Additionally, the adversary will be able to bypass Multi-Factor Authentication (MFA) protections.

Chapter 3: The impact of the attack:

As said earlier there were 300,000 customers of SolarWinds at the time of the attack. However, only 18000 of these had installed the compromised updates.

Of the victims, around 20 percent were US government institutions and agencies such as the Department of Homeland Security, the State Department, the National Nuclear Security Administration, and the Department of Energy, among many others.

The remaining 80 percent of victims were private corporations, but they were big players in their industry with their fair share of high-profile clients. The hack affected companies like Cisco, Intel, Deloitte, and Microsoft, as well as some medical institutions, hospitals, and universities.

In addition to the theft of data, the attack caused costly inconvenience to tens of thousands of SolarWinds customers, who had to check whether they had been breached, and had to take systems offline and begin months-long decontamination procedures as a precaution. U.S. Senator Richard J. Durbin described the cyberattack as tantamount to a declaration of war.

A survey asked about the financial impact of the attack and found that the average impact was 11% of annual revenue or about $12 million per company. Companies in the U.S. reported an average of a 14% impact on annual revenue with the averages in the U.K. and Singapore at 8.6% and 9.1% respectively.

According to the BitSight-Kovrr analysis of the SolarWinds attack, the estimated losses was said to be $90,000,000, which includes incident response and forensic services for companies who were impacted by this incident and have cyber insurance coverage.

In the SolarWinds attack, while a specific technology was targeted that has a significant customer, it appears the threat actor has avoided large scale exploitation of organizations. This means that even though the hackers could cause immense losses of over 200 million dollars, they did not do so.

Around January 5, 2021, SolarWinds investors filed a class action lawsuit against the company in relation to its security failures and subsequent fall in share price.

While roughly 80% of these customers are located in the United States, this work so far has also identified victims in seven additional countries. This includes Canada and

Mexico in North America; Belgium, Spain and the United Kingdom in Europe; and Israel and the UAE in the Middle East. It's certain that the number and location of victims will keep growing.

Additional analysis sheds added light on the breadth of these attacks. The initial list of victims includes not only government agencies, but security and other technology firms as well as non-governmental organizations, as shown in the chart below.

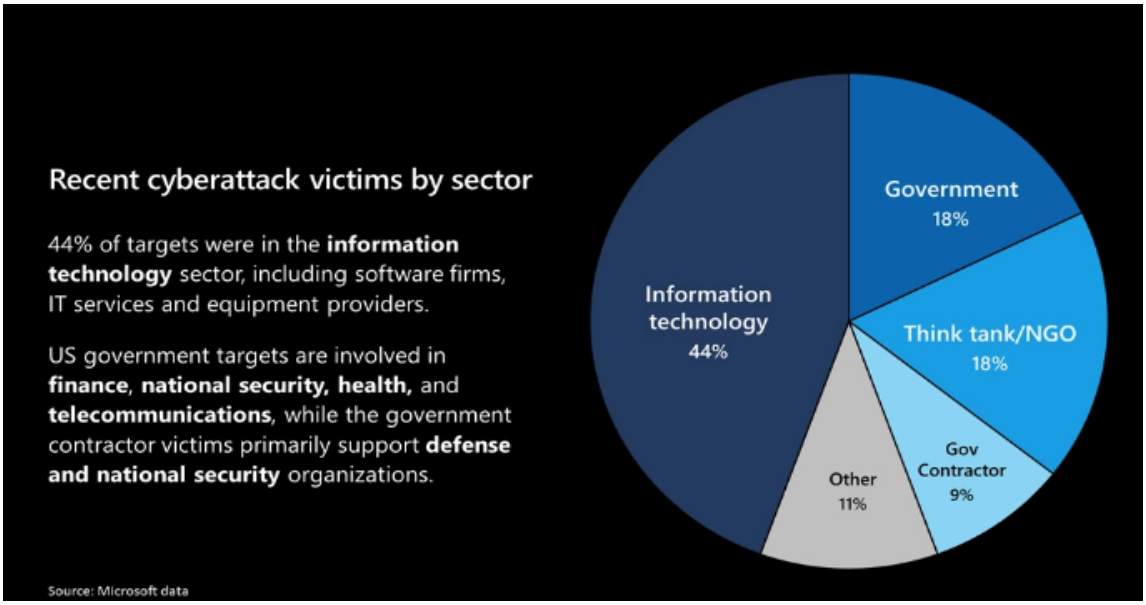

Fig. 7: Chart of impacted organizations

4. Security Measures:

CISA (Cybersecurity and Infrastructure Security Agency) recommends Category organizations to:

- Maintain up-to-date antivirus signatures and engines. See Protecting Against Malicious Code.
- Ensure systems have the latest security updates. See Understanding Patches and Software Updates.
- Enforce a strong password policy. See Choosing and Protecting Passwords.
- Exercise caution when opening email attachments, even if the attachment is expected and the sender appears to be known. See Using Caution with Email Attachments.
- Sign up to receive CISA's alerts on security topics and threats.
- Sign up for CISA's free vulnerability scanning and testing services to help organizations secure internet-facing systems from weak configuration and known vulnerabilities.
- Email vulnerability@cisa.dhs.gov to sign up. See https://www.cisa.gov/cyber-resource-hub for more information about vulnerability scanning and other CISA cybersecurity assessment services.
- Other organizations should continue enhanced monitoring for any possible follow-on adversary activity.

On December 8, 2020, before other organizations were known to have been breached, FireEye published countermeasures against the red team tools that had been stolen from FireEye.

On December 15, 2020, Microsoft announced that SUNBURST, which only affects Windows platforms, had been added to Microsoft's malware database and would, from December 16 onwards, be detected and quarantined by Microsoft Defender.

GoDaddy handed ownership to Microsoft of a command-and-control domain used in the attack, allowing Microsoft to activate a killswitch in the SUNBURST malware, and to discover which SolarWinds customers were infected.

On December 14, 2020, the CEOs of several American utility companies convened to discuss the risks posed to the power grid by the attacks.[1] On December 22, 2020, the North American Electric Reliability Corporation asked electricity companies to report their level of exposure to SolarWinds software.

SolarWinds unpublished its featured customer list after the hack.

5. Suggested Security Measures:

It is still not clear how the hackers were able to plant sunspot malware in the Orion platform. However, fingers are being pointed out at an intern at SolarWinds who kept a weak password “solarwinds123” for a server and this password was available on GitHub. This may not be the actual cause for the access. But this suggests several security measures that need to be taken up as a result of this attack:

- Regularly change passwords
- Always keep strong passwords
- Never disclose information regarding the Government without consultation
- Assign Cybersecurity Officers to look after the Cyber Security Scenario within an organization
- Regularly update softwares since they contain patches and bug fixes.
- Reduce unnecessary privileges given for performing a task. This prevents hackers from gaining privileged access.
- Maintain logs of all operations in the system.
- Always have a contingency plan in place to tackle the situation should it happen again

6. References:

- https://www.splunk.com/en_us/blog/security/a-golden-saml-journey-solarwinds-continued.html
- https://msrc.microsoft.com/blog/2020/12/december-21st-2020-solorigate-resource-center/
- https://gist.github.com/fr0gger/46b0998cd9c4d7a2ba7a81fbe4f9e2b3
- https://www.mandiant.com/resources/blog/unc2452-merged-into-apt29
- https://www.youtube.com/watch?v=8O8v022f06g&t=2002s
- https://www.youtube.com/watch?v=Kf7Motm36Go
- https://www.secureworld.io/industry-news/microsoft-reports-techniques-solarwinds
- https://blogs.microsoft.com/on-the-issues/2020/12/17/cyberattacks-cybersecurity-solarwinds-fireeye/