Headline

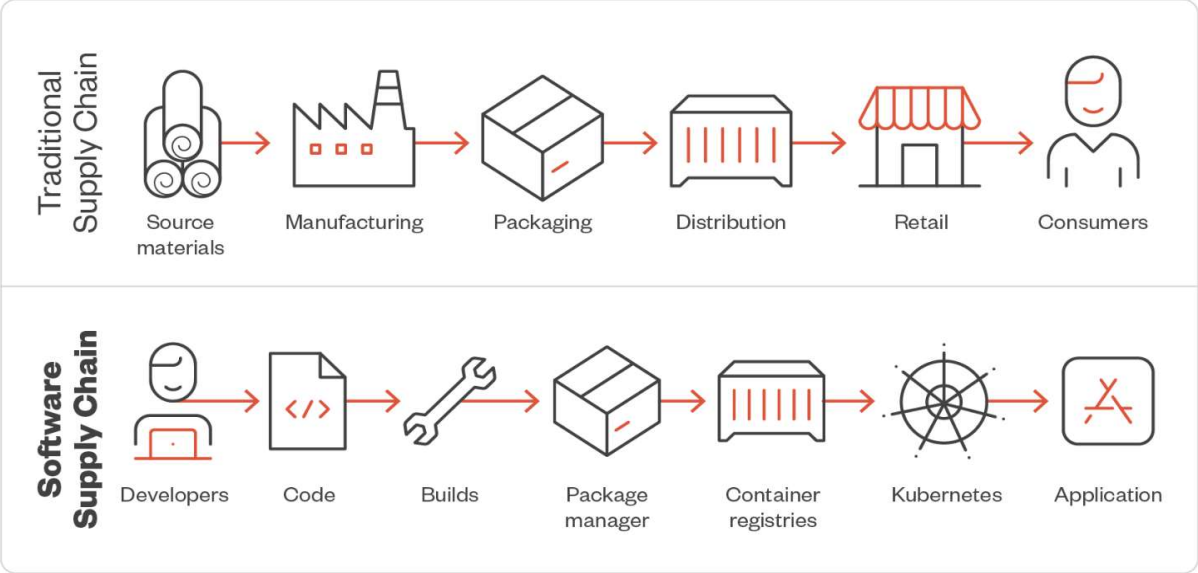
SW Supply Chain Security Threats and Countermeasures

Young-shik Song, Senior Manager / Hi-Tech Operation Team

■ Overview

Using open source code to develop competitive software has become a natural phenomenon. This is because unnecessary rework can be avoided and development can be carried out efficiently by using existing code and open source libraries such as Log4j. It allows for a reduction in development costs and the development of higher-quality code, thereby enhancing technological competitiveness. However, worryingly, 'software supply chain attacks' that exploit this phenomenon have recently become common and are compromising corporate networks.

A software supply chain attack is a threat that occurs when an attacker maliciously intervenes in the software development or distribution process. The attacker's aim is to infiltrate user systems by inserting malicious code into trusted software.

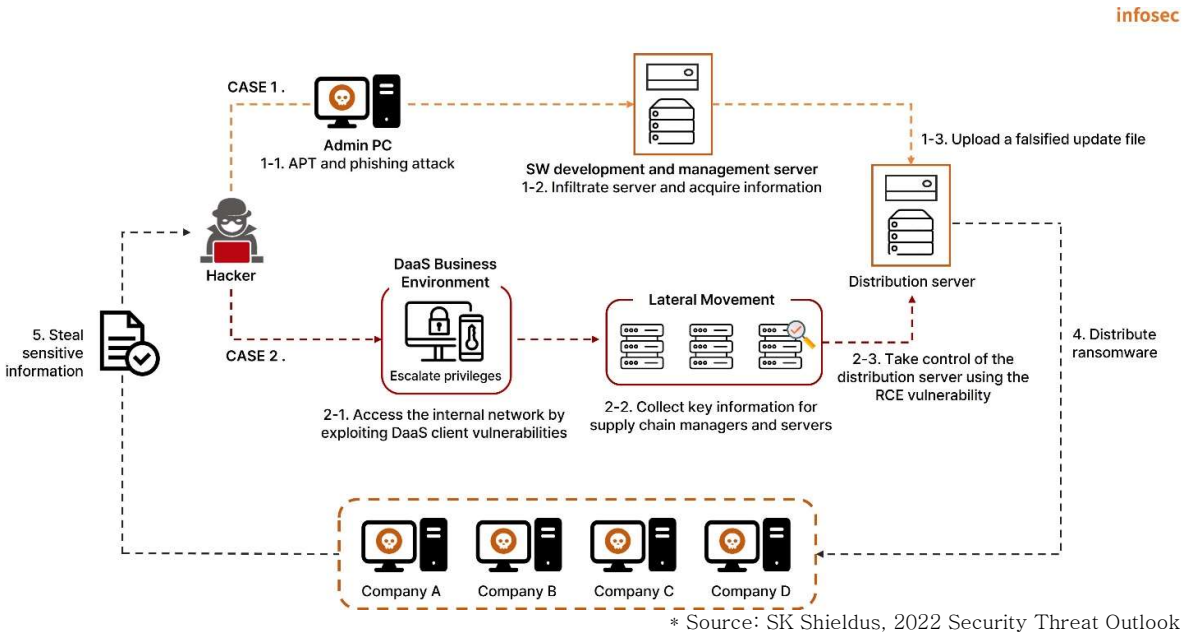


* Source: Trendmicro

Figure 1. Structural comparison between a typical supply chain and a software supply chain

■ Attack Types and Cases

Supply chain attacks can cause significant damage in a short period of time, and rank among the major attacks that occur every year in terms of frequency of occurrence. Hackers typically first attack software vendors through advanced persistent threat (APT) and phishing attacks, and then modify software distribution server files on the internal network. In addition, attackers attack the supply chain in a variety of ways, including by discovering and exploiting open source/library vulnerabilities.



* Source: SK Shieldus, 2022 Security Threat Outlook

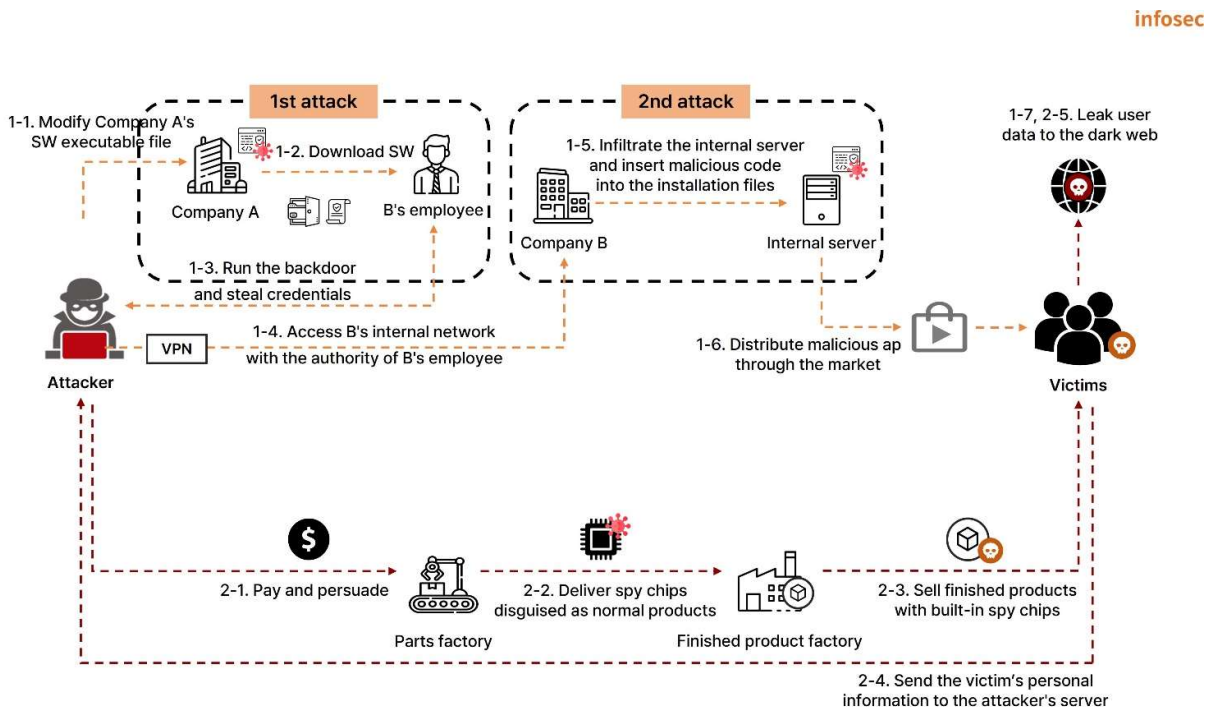
Figure 2. Supply chain attack method

1. Software vendor attack cases

Attackers infiltrate the systems of software developers or vendors and then inject malicious code into the software they provide. Users think they are downloading software from a trusted source, but in reality it is infected software. This allows the attackers to abuse the vendor's authority in order to steal data from customers and partner organizations, spread malware and perform attacks, resulting in ransomware infections and information leakage.

In 2021, the cybercrime group REvil attacked the US IT company Kaseya by exploiting vulnerabilities in software used in the company's remote monitoring and management solutions. REvil then used the stolen credentials to deploy ransomware to hundreds of customers.

A hacker group suspected to be North Korea's Lazarus has launched a series of supply chain attacks on the software company 3CX. Rather than targeting a single company, this group of hackers spread their attacks by targeting companies that use 3CX's products and services or that have a network connection with 3CX.



* Source: EQST, Security Trend Report for the First Half of 2023

Figure 3. Confirmed supply chain attack scenario

The 3CX supply chain attack is an example of a primary software (X_Trader) supply chain attack leading to a secondary software (3CX) supply chain attack.

An employee of 3CX downloaded a malware-infected program called X_Trader from the software provider Trading Technologies, and this program infected the employee's PC.

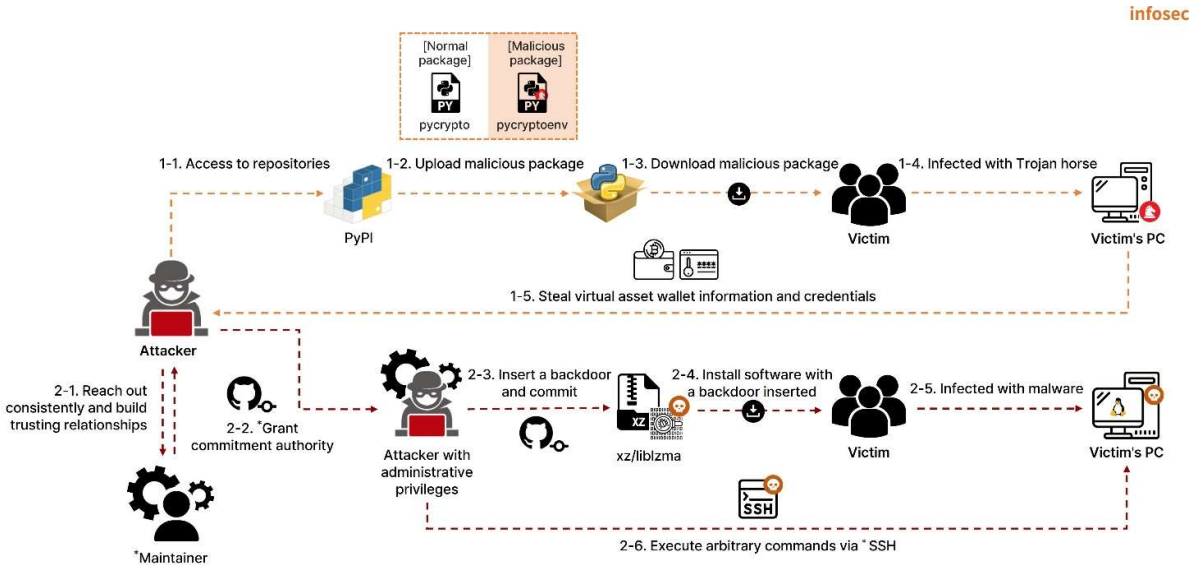
The attackers thus gained access to a 3CX employee's PC. They then exploited the employee's credentials to infiltrate the 3CX build server, and inserted malware into the 3CX software. The falsified software was distributed to countries around the world as an installation file through the official website.

In the first attack, VEILEDSIGNAL, which is known to be a backdoor used by the North Korean hacking group Lazarus, was discovered in the infected X_Trader. In the second attack, the 3CX supply chain attack, Gopuram malware was discovered. Based on this, it is presumed that North Korea's Lazarus is behind the attack.

2. Open source/library attack cases

In recent years, a growing number of companies have been leveraging open source (or publicly accessible) code to maximize the efficiency of software development. However, when vulnerabilities are discovered in code, the organizations using that code are exposed to great risk. In addition to exploiting already known vulnerabilities, attackers can also attempt to spread malware by inserting malicious code into packages.

A typical example of an open source attack is a supply chain attack through the Python Package Index (PyPI) community.



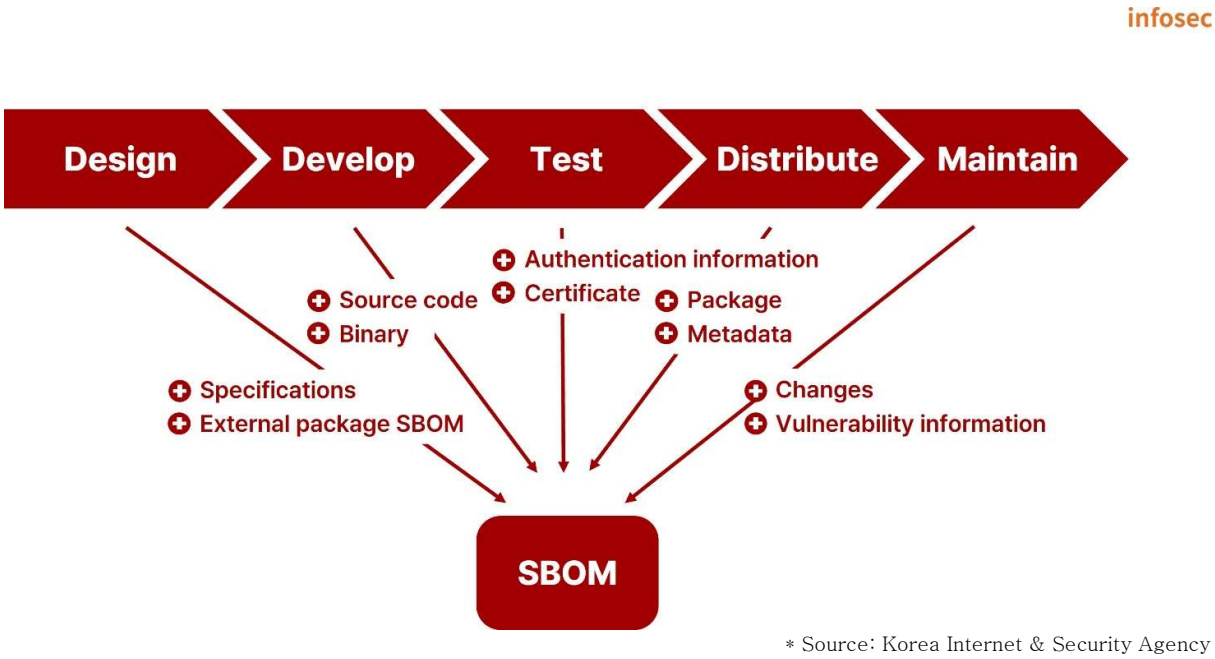
* Source: SK Shieldus, Security Trends for the First Half of 2024

Figure 4. Supply chain attack through the Python Package Index (PyPI) Community

The attacker accesses PyPI, a Python repository, inserts a Trojan horse into a malicious package with a name similar to that of a normal package, and uploads it. When a victim downloads the malicious package, the Trojan horse is executed and the attacker can steal virtual asset wallet information and credentials from the victim's PC. In this type of attack, the attacker uses typosquatting techniques by using package names such as 'pycryptoenv' or 'pycryptoconf,' which are similar to the popular Python library 'pycrypto,' or exploits user typos such as 'pycryipto.' For more details, please refer to SK Shieldus' First Half Security Trends.

Software-based attacks are very widespread, with 66% of all attacks targeting vendor code. However, supply chain attacks require attention due to the various forms. For example, these attacks can damage microchips, laptops, Internet of things (IoT) devices, and operational technology (OT), and can even target firmware, which is software embedded in hardware.

■ Countermeasures



* Source: Korea Internet & Security Agency

Figure 5. Systematic management method for software bills of materials (SBOMs)

1. Manage Software Bills of Materials (SBOMs)

The first step to securing a software supply chain is to identify the software components. It is important to manage the software bills of materials (SBOM), which include information on the commercial and open source software components of a product. By managing SBOMs systematically, it is possible to immediately check and take action on newly discovered vulnerabilities. For this reason, the United States and Europe have made SBOM submission mandatory to strengthen software supply chain security.

2. Check the supplied software

It is necessary to inspect software supplied from outside, specifically whether the software is provided through official channels and is code-signed. After installing/updating software, it's a good idea analyze any unusual behavior on the PC, such as attempts to connect to the outside.

3. Strengthen the vulnerability response

The point at which supply chain attacks occur is in operational software. In most cases, vulnerabilities in the supplied software are eliminated during the development, testing, and distribution processes, but new vulnerabilities may be found during the operation process. Operations organizations must analyze these vulnerabilities to determine their actual impact and take immediate action for high-risk vulnerabilities. For relatively low-risk vulnerabilities, appropriate measures should be taken considering the interruption of system service.

■ Conclusion

To address bugs and security issues, most software vendors provide updates through a central server for maintenance purposes. In this software supply chain ecosystem, attackers infiltrate suppliers' networks and then alter outgoing updates or insert malicious code. Afterwards, they gain control over the normal functions of the software and continue their attacks, such as ransomware and information leaks.

Supply chain attacks are more dangerous because they can spread beyond a single corporate target to other companies that use the company's products or have a network connection. SK Shieldus' 2024 EQST Annual Report labelled such software supply chain attacks as one of the top five cyber threats expected in 2024. With the ongoing Russian-Ukrainian war and the Israeli-Palestinian conflict raging this year, we can expect continued attacks on new supply chains targeting businesses and critical global infrastructure.

SK Shieldus provides consulting on establishing an open source SW management system. Please visit the SK Shieldus website for details.