

Malicious NuGet Packages Used to Target .NET Developers

A newly discovered attack has been targeting .NET developers with malicious packages loaded to the NuGet repository, JFrog's security researchers explain.

A package manager helping developers share and consume reusable code, NuGet allows developers to create code packages using the NuGet client app and have them published in either public or private repositories.

While attacks abusing open source repositories are nothing new, NuGet has not seen severe malicious activity until now, aside from packages designed to spread phishing links.

This incident, however, brings NuGet in line with NPM and PyPI repositories, which are often targeted by cybercriminals: malicious packages submitted to the repository contained code that triggered the download of a second-stage payload.

As part of the attack, typosquatting was used to trick developers into downloading the malicious packages. With roughly 150,000 downloads racked up before the packages were removed from NuGet, the operation was highly successful.

Within the malicious packages, the attackers included a PowerShell script that was executed at the package's installation. The script changed the system configuration so that PowerShell scripts could be executed without restrictions and then fetched a secondary payload from a remote server.

The payload, a Windows executable file, was designed to steal cryptocurrency, extract and execute code from Electron archives, and drop a small updater executable that ensures the

malware is always up-to-date.

To carry out the attack, the threat actors abused a feature in older Visual Studio versions where scripts could be placed in the 'tools' directory of a NuGet package to have them executed automatically with no constraints on specific events.

Newer iterations of Visual Studio ignore run-on-install scripts, but they still honor the execution of scripts added with older versions, without displaying a warning during the installation of a NuGet package.

"These kinds of autorun mechanisms are a big reason why we can find thousands of malicious packages plaguing the NPM and PyPI ecosystems as compared to the Go package ecosystem, for example, in which the client will not cause code to automatically run when a module is installed. The attacker's code is much more likely to run if only package installation is needed to trigger it," JFrog points out.

As part of the observed attack, the security researchers identified 13 NuGet packages containing the same malicious payload, most of which were impersonating popular packages.

The most popular of these, Coinbase.Core, had over 120,000 downloads at the time it was removed from the repository. The second most popular, Anarchy.Wrapper.Net, had over 30,000 downloads, while the third, DiscordRichPresence.API, had over 14,000.

"The top three packages were downloaded an incredible amount of times – this could be an indicator that the attack was highly successful, infecting a large amount of machines. However, this is not a fully reliable indicator of the attack's success since the attackers could have automatically inflated the download count (with bots) to make the packages seem more legitimate," JFrog notes.

Related: Top 10 Security, Operational Risks From Open Source

Code

Related: Vulnerability in Popular JsonWebToken Open Source Project Leads to Code Execution

Related: Google Announces Vulnerability Scanner for Open Source Developers

<https://www.securityweek.com/malicious-nuget-packages-used-to-target-net-developers/>