# CVE Detail – CVE-2022-36069

Poetry is a dependency manager for Python. When handling dependencies that come from a Git repository instead of a registry, Poetry uses various commands, such as `git clone`. These commands are constructed using user input (e.g. the repository URL). When building the commands, Poetry correctly avoids Command Injection vulnerabilities by passing an array of arguments instead of a command string. However, there is the possibility that a user input starts with a dash (`-`) and is therefore treated as an optional argument instead of a positional one. This can lead to Code Execution because some of the commands have options that can be leveraged to run arbitrary executables. If a developer is exploited, the attacker could steal credentials or persist their access. If the exploit happens on a server, the attackers could use their access to attack other internal systems. Since this vulnerability requires a fair amount of user interaction, it is not as dangerous as a remotely exploitable one. However, it still puts developers at risk when dealing with untrusted files in a way they think is safe, because the exploit still works when the victim tries to make sure nothing can happen, e.g. by vetting any Git or Poetry config files that might be present in the directory. Versions 1.1.9 and 1.2.0b1 contain patches for this issue.

## Threat-Mapped Scoring

Score: 1.8

Priority: P4 - Informational (Low)

## EPSS

EPSS Score: N/A

Percentile: 0.65337

## CVSS Scoring

CVSS v3.1 Score: 7.3

Severity: HIGH

## Mapped CWE(s)

* CWE-88: Improper Neutralization of Argument Delimiters in a Command ('Argument Injection')
* CWE-94: Improper Control of Generation of Code ('Code Injection')

## CAPEC(s)

* CAPEC-137: Parameter Injection
* CAPEC-174: Flash Parameter Injection
* CAPEC-242: Code Injection
* CAPEC-35: Leverage Executable Code in Non-Executable Files
* CAPEC-41: Using Meta-characters in E-mail Headers to Inject Malicious Payloads
* CAPEC-460: HTTP Parameter Pollution (HPP)
* CAPEC-77: Manipulating User-Controlled Variables
* CAPEC-88: OS Command Injection

## ATT&CK Techniques

* T1027.009: Embedded Payloads
* T1564.009: Resource Forking
* T1027.006: HTML Smuggling

## Used By (Actors/Tools)

* Pikabot (malware)
* macOS.OSAMiner (malware)
* EnvyScout (malware)
* Emotet (malware)
* DUSTTRAP (malware)
* BADHATCH (malware)
* DUSTPAN (malware)
* Moneybird (malware)
* Keydnap (malware)
* IcedID (malware)
* CHIMNEYSWEEP (malware)
* MultiLayer Wiper (malware)
* Netwalker (malware)
* SMOKEDHAM (malware)
* Uroburos (malware)
* DEADEYE (malware)
* ComRAT (malware)
* QakBot (malware)
* OSX/Shlayer (malware)
* DEADWOOD (malware)
* Dtrack (malware)
* Invoke-PSImage (tool)
* Lazarus Group (intrusion-set)
* TA577 (intrusion-set)
* APT29 (intrusion-set)
* C0021 (campaign)
* Moonstone Sleet (intrusion-set)

## Affected Products

* cpe:2.3:a:python-poetry:poetry:\*:\*:\*:\*:\*:python:\*:\*
* cpe:2.3:a:python-poetry:poetry:1.2.0:alpha1:\*:\*:\*:python:\*:\*
* cpe:2.3:a:python-poetry:poetry:1.2.0:alpha2:\*:\*:\*:python:\*:\*