

Compromising Developers with Malicious Extensions

VS Code, Cursor AI, and the Backdoor You Didn't See Coming

```
Merlin[listeners][c8ef72d7-1b0f-4a84-b10b-3391ea7f77ba]»
[+] 2025-04-14T21:22:30Z New authenticated Agent checkin for 29b7a59a-eee0-478e-8df7-c0e012b5f1cc at 2025-04-14T21:22:30Z

[-] 2025-04-14T21:23:02Z Results of job CjyQz0cxW for agent 29b7a59a-eee0-478e-8df7-c0e012b5f1cc at 2025-04-14T21:23:02Z
    Configuration data received for Agent 29b7a59a-eee0-478e-8df7-c0e012b5f1cc and updated. Issue the "info" command to view it.

Merlin[listeners][c8ef72d7-1b0f-4a84-b10b-3391ea7f77ba]»
Merlin[listeners][c8ef72d7-1b0f-4a84-b10b-3391ea7f77ba]»
```

Mazin Ahmed

BLACKHAT MEA 2025 | Riyadh, Saudi Arabia 

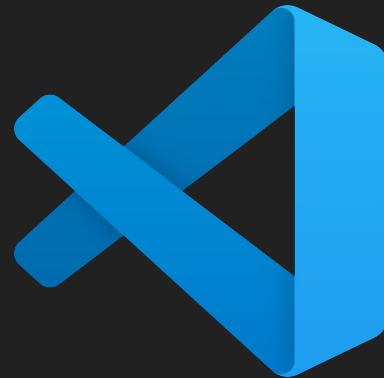
WHO AM I?

Mazin Ahmed

- **AppSec Engineering, Security R&D in a Financial Institution**
- **Founder of FullHunt**
- **Occasional Bug Bounty Hunter: acknowledged by Meta, Twitter(X), LinkedIn, Zoom, U.S. Department of Defense, and more**
- **In love ❤️ with AI Security, Cloud security, security automation, DevSecOps, distributed systems, and Web-App security**

AI-powered IDEs

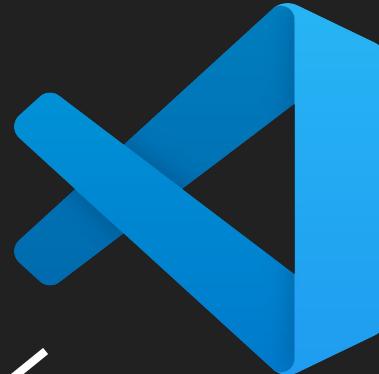
W Windsurf



AI-powered IDEs



AI-powered IDEs



W Windsurf



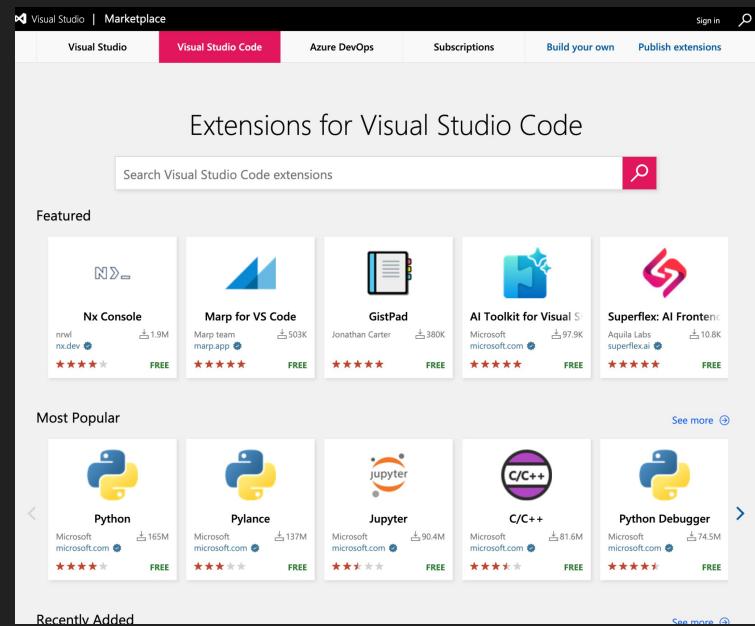
CURSOR

VS Code Marketplace

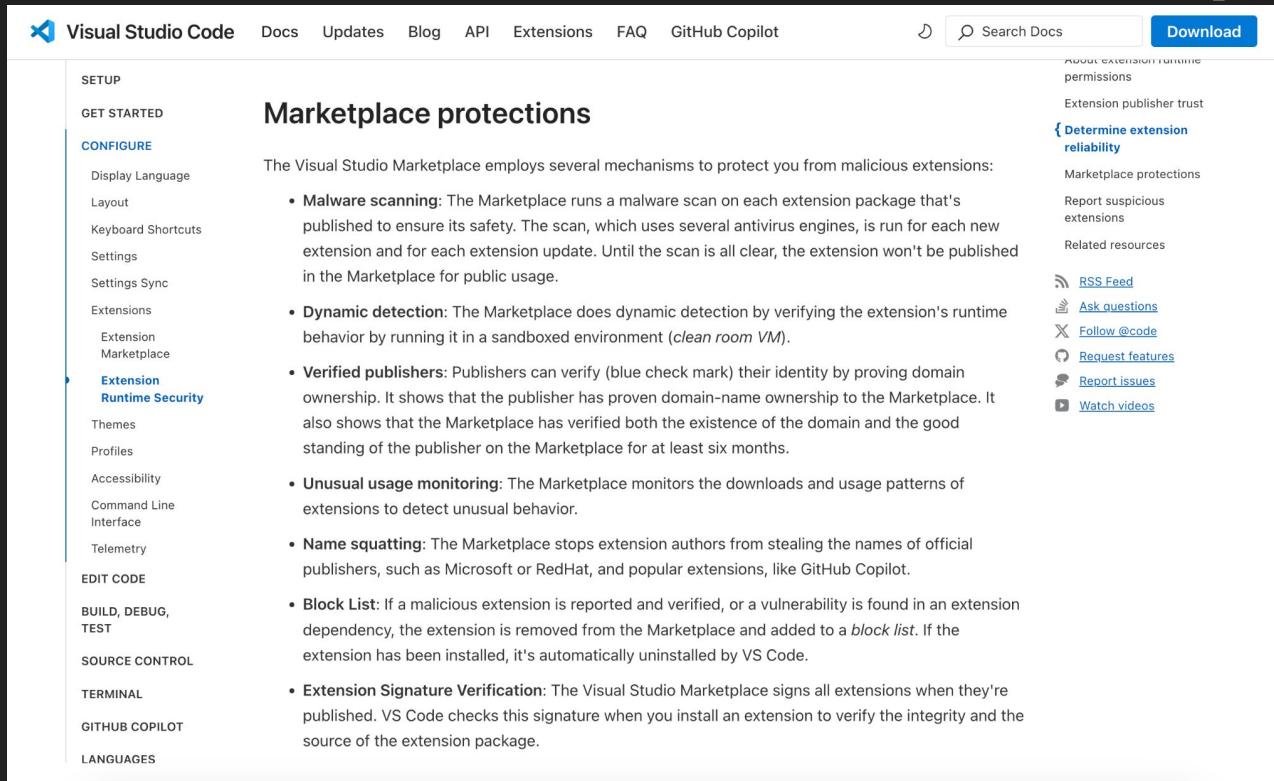
VS Code Marketplace

VSCode Marketplace is owned by Microsoft.

Publishing an extension goes through iterations of quality and **security** checks.



How safe is the Visual Studio Code Marketplace?



The screenshot shows a white sidebar on the left with a dark background containing a navigation menu. The menu items are: SETUP, GET STARTED, CONFIGURE, EXTENSION RUNTIME SECURITY (which is highlighted in blue), EDIT CODE, BUILD, DEBUG, TEST, SOURCE CONTROL, TERMINAL, GITHUB COPILOT, and LANGUAGES. To the right of the sidebar is the main content area. At the top of the content area is a navigation bar with links for Visual Studio Code, Docs, Updates, Blog, API, Extensions, FAQ, GitHub Copilot, a search bar labeled "Search Docs", and a "Download" button. Below the navigation bar is a section titled "Marketplace protections". A sub-section header "ADDITIONAL EXTENSION PUBLISHING PERMISSIONS" is visible above a list of links. The main content area contains a paragraph about marketplace protections followed by a bulleted list of safety mechanisms.

The Visual Studio Marketplace employs several mechanisms to protect you from malicious extensions:

- **Malware scanning:** The Marketplace runs a malware scan on each extension package that's published to ensure its safety. The scan, which uses several antivirus engines, is run for each new extension and for each extension update. Until the scan is all clear, the extension won't be published in the Marketplace for public usage.
- **Dynamic detection:** The Marketplace does dynamic detection by verifying the extension's runtime behavior by running it in a sandboxed environment (*clean room VM*).
- **Verified publishers:** Publishers can verify (blue check mark) their identity by proving domain ownership. It shows that the publisher has proven domain-name ownership to the Marketplace. It also shows that the Marketplace has verified both the existence of the domain and the good standing of the publisher on the Marketplace for at least six months.
- **Unusual usage monitoring:** The Marketplace monitors the downloads and usage patterns of extensions to detect unusual behavior.
- **Name squatting:** The Marketplace stops extension authors from stealing the names of official publishers, such as Microsoft or RedHat, and popular extensions, like GitHub Copilot.
- **Block List:** If a malicious extension is reported and verified, or a vulnerability is found in an extension dependency, the extension is removed from the Marketplace and added to a *block list*. If the extension has been installed, it's automatically uninstalled by VS Code.
- **Extension Signature Verification:** The Visual Studio Marketplace signs all extensions when they're published. VS Code checks this signature when you install an extension to verify the integrity and the source of the extension package.

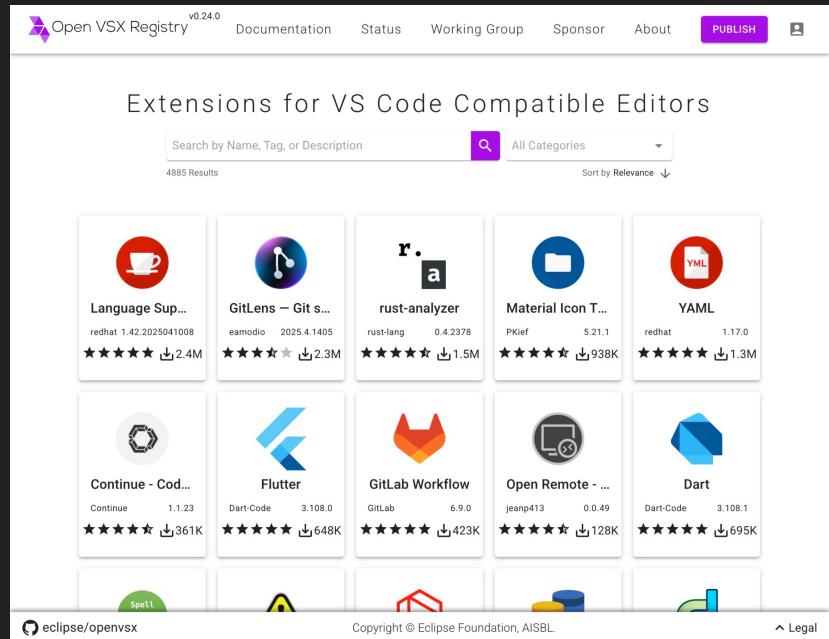
<https://code.visualstudio.com/docs/configure/extensions/extension-runtime-security>

OpenVSX Marketplace

Marketplace used by Cursor AI, Windsurf, AWS Kiro, and other IDEs

OpenVSX Marketplace

- Verified Publishers and Namespaces
- Mandatory Licensing and Publisher Agreement
- Abuse Reporting Mechanism
- Transparent Governance



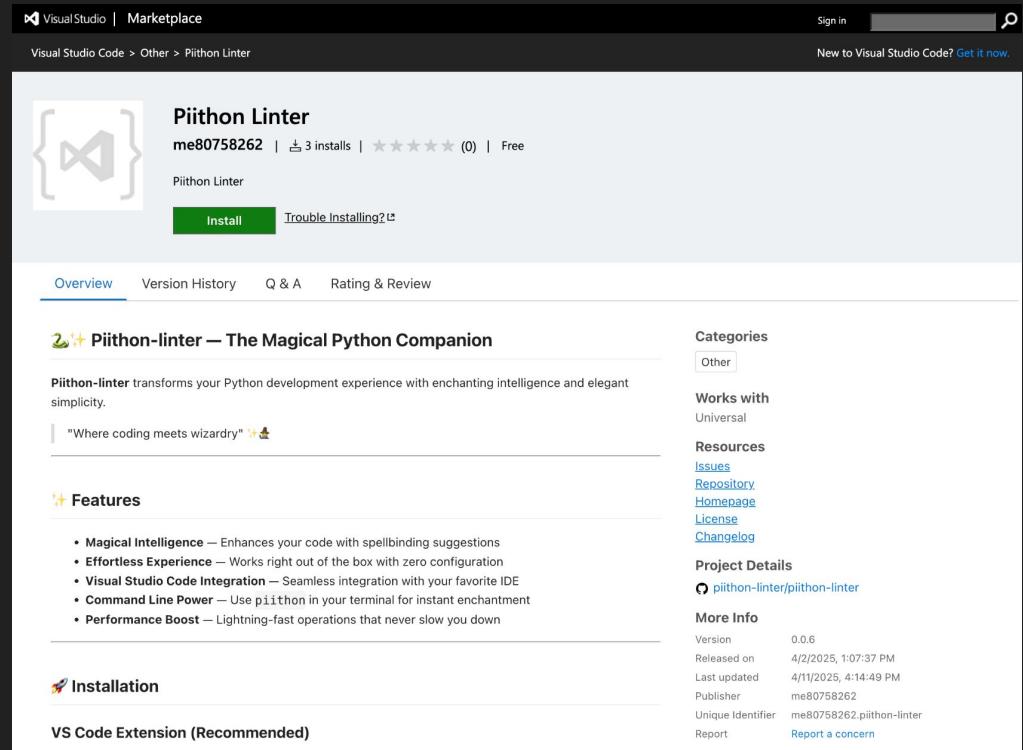
Compliance != Security

Developing a malicious VS Code Extension

Piithon-linter: A malicious Python linter/formatter

I built piithon-linter

A malicious Python linter/formatter with “magical” capabilities to format your code.



The screenshot shows the Visual Studio Marketplace page for the extension "Piithon Linter". The page includes the following details:

- Publisher:** me80758262
- Installs:** 3 installs
- Rating:** 0 stars (0 reviews)
- Category:** Other
- Works with:** Universal
- Resources:** Issues, Repository, Homepage, License, Changelog
- Project Details:** piithon-linter/piithon-linter
- More Info:** Version 0.0.6, Released on 4/2/2025, Last updated 4/11/2025, Publisher me80758262, Unique Identifier me80758262.piithon-linter, Report a concern

The main content area features a heading "Piithon Linter — The Magical Python Companion" with a subtext "Piithon-linter transforms your Python development experience with enchanting intelligence and elegant simplicity." Below this is a quote "Where coding meets wizardry!" and a section titled "Features" listing:

- Magical Intelligence — Enhances your code with spellbinding suggestions
- Effortless Experience — Works right out of the box with zero configuration
- Visual Studio Code Integration — Seamless integration with your favorite IDE
- Command Line Power — Use piithon in your terminal for instant enchantment
- Performance Boost — Lightning-fast operations that never slow you down

At the bottom, there's a "Installation" section with a link "VS Code Extension (Recommended)".





Piithon Linter

me80758262 | 3 installs | ★★★★★ (0) | Free

Piithon Linter

[Install](#) [Trouble Installing?](#)

[Overview](#) [Version History](#) [Q & A](#) [Rating & Review](#)

🐍 Piithon-linter — The Magical Python Companion

Piithon-linter transforms your Python development experience with enchanting intelligence and elegant simplicity.

"Where coding meets wizardry" 🧙‍♂️

✨ Features

- Magical Intelligence — Enhances your code with spellbinding suggestions
- Effortless Experience — Works right out of the box with zero configuration
- Visual Studio Code Integration — Seamless integration with your favorite IDE
- Command Line Power — Use piithon in your terminal for instant enchantment
- Performance Boost — Lightning-fast operations that never slow you down

🚀 Installation

VS Code Extension (Recommended)

Categories
Other

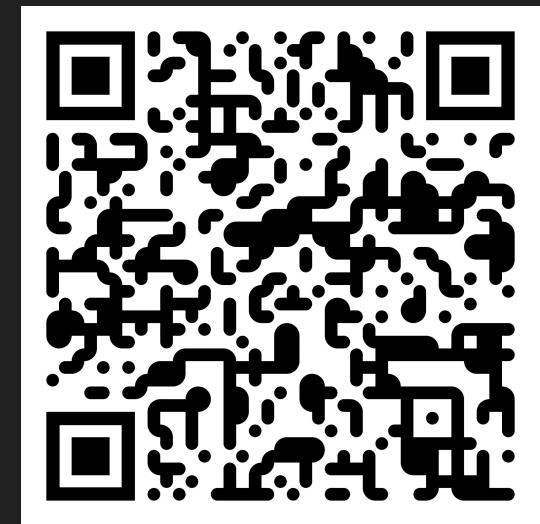
Works with
Universal

Resources
[Issues](#)
[Repository](#)
[Homepage](#)
[License](#)
[Changelog](#)

Project Details
[piithon-linter/piithon-linter](#)

More Info

Version	0.6
Released on	4/2/2025, 1:07:37 PM
Last updated	4/11/2025, 4:14:49 PM
Publisher	me80758262
Unique Identifier	me80758262.piithon-linter
Report	Report a concern



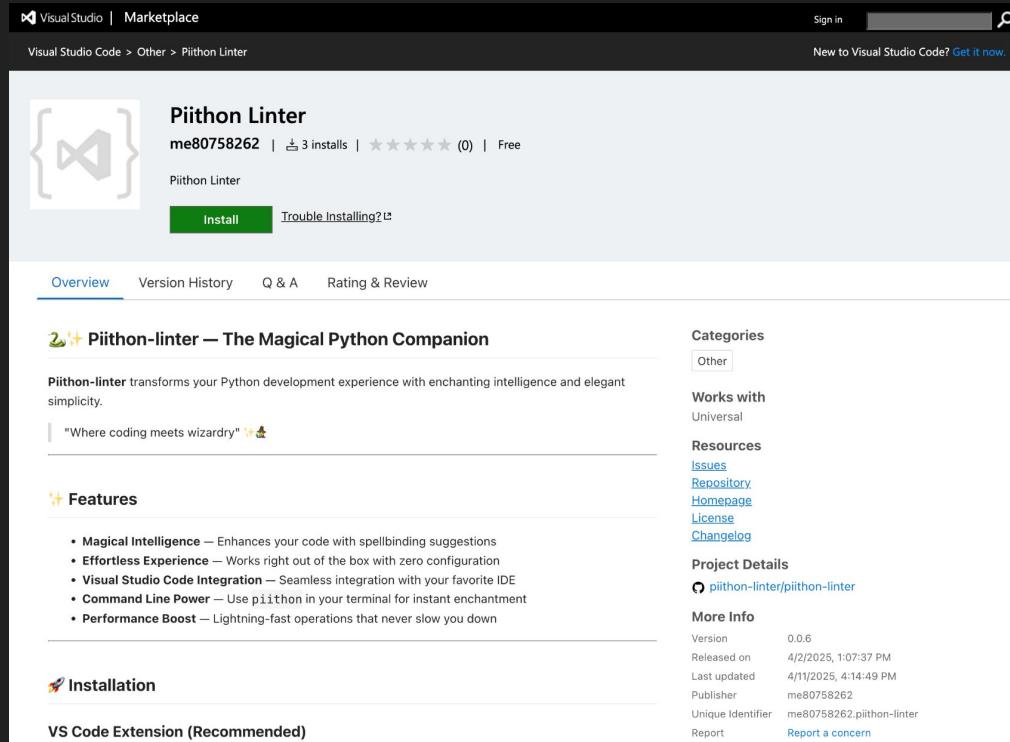
Want to install this extension?

Piithon-linter: A malicious Python linter/formatter

First iteration

Every time VS Code starts, the malicious VS Code extension exfiltrates the developer's environment variables and metadata to a C2 server.

It does not actually have any formatting/linting capabilities.



The screenshot shows the Visual Studio Marketplace page for the "Piithon Linter" extension. The extension icon is a grey square containing a white Python logo. The title is "Piithon Linter" by user "me80758262". It has 3 installs and no reviews. The status is "Free". There are "Install" and "Trouble Installing?" buttons. Below the main header, there are tabs for "Overview", "Version History", "Q & A", and "Rating & Review". The "Overview" tab is selected. The description reads: "Piithon-linter transforms your Python development experience with enchanting intelligence and elegant simplicity." A quote below says, "'Where coding meets wizardry' 🧙‍♂️". The "Features" section lists: • Magical Intelligence – Enhances your code with spellbinding suggestions • Effortless Experience – Works right out of the box with zero configuration • Visual Studio Code Integration – Seamless integration with your favorite IDE • Command Line Power – Use piithon in your terminal for instant enchantment • Performance Boost – Lightning-fast operations that never slow you down. The "Installation" section indicates it is a "VS Code Extension (Recommended)". On the right side, there are sections for "Categories" (Other), "Works with" (Universal), "Resources" (Issues, Repository, Homepage, License, Changelog), "Project Details" (@piithon-linter/piithon-linter), and "More Info" (Version 0.0.6, Released on 4/2/2025, Last updated 4/11/2025, Publisher me80758262, Unique Identifier me80758262.piithon-linter, Report a concern).

Piithon-linter: A malicious Python linter/formatter

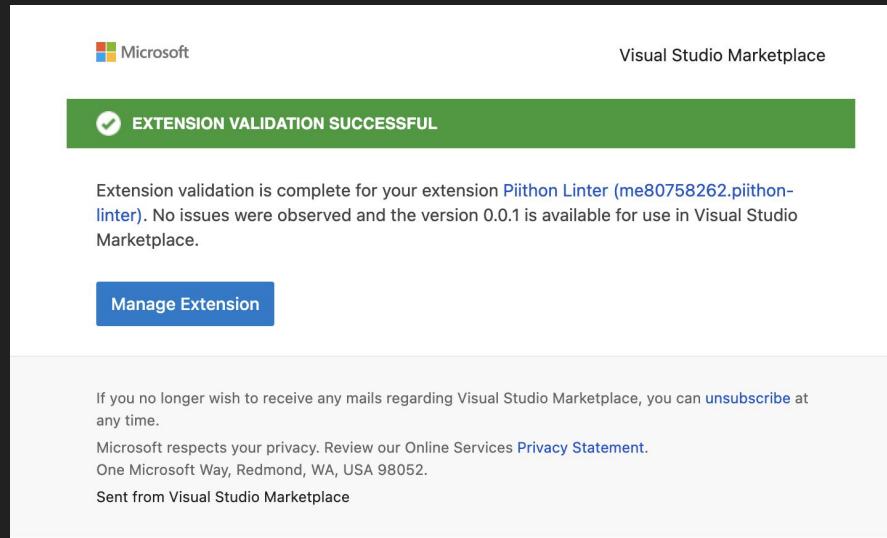
First iteration

Every time VS Code starts, the malicious VS Code extension exfiltrates the developer's environment variables and metadata to a C2 server.

It does not actually have any formatting/linting capabilities.

```
=BEGIN  
GIT_EMAIL=developer@acme.org  
IP_ADDRESS=1.2.3.4  
HOSTNAME=INV5966  
ENVs:  
ALLUSERSPROFILE=C:\P***Data  
APPDATA=C:\U***ming  
COMPUTERNAME=INV5***5966  
ComSpec=C:\W***.exe  
CommonProgramFiles=C:\P***iles  
CommonProgramFiles(x86)=C:\P***iles  
CommonProgramW6432=C:\P***iles  
DOTNET_CLI_TELEMETRY_OPTOUT=true***  
DriverData=C:\W***Data
```

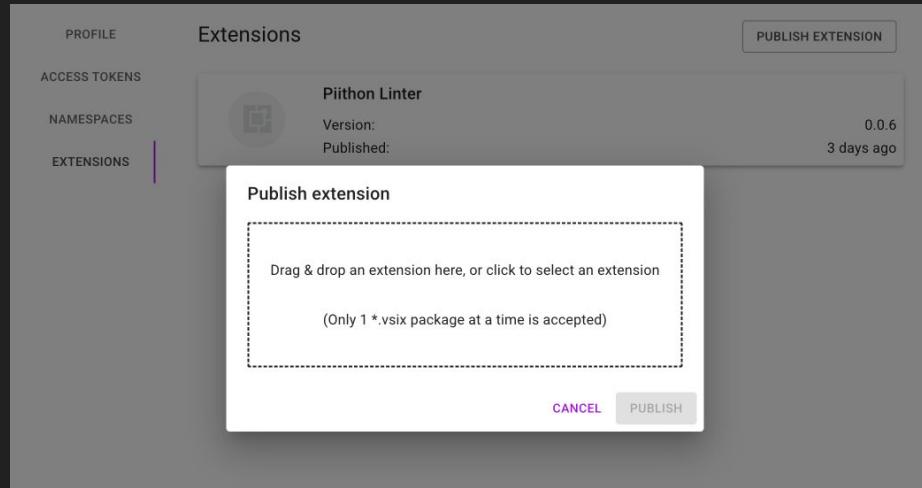
Piithon-linter: A malicious Python linter/formatter



Microsoft Marketplace accepted the initial version.

The build was not flagged.

Piithon-linter: A malicious Python linter/formatter



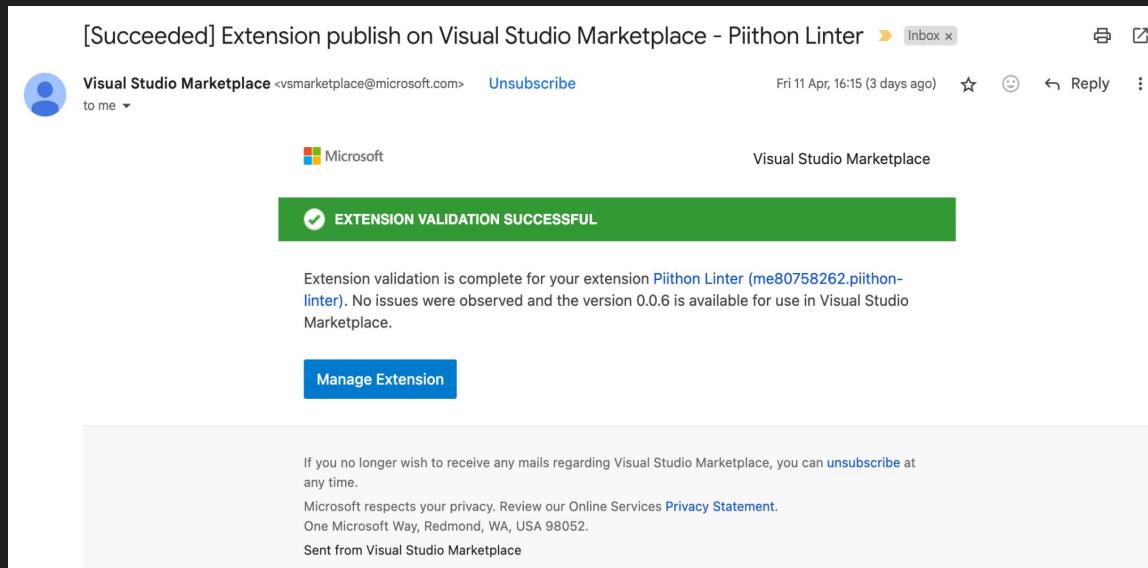
Open-VSX also accepted the malicious extension.

Piithon-linter: A malicious Python linter/formatter

The second Piithon-linter had more realistic offensive capabilities:

- **Sends Environment Variables to a C2 server.**
- **Runs detection checks to check whether an endpoint security solution is installed.**
- **Check if the developer is from a particular country (e.g., US).**
- **Deploy a Merlin agent depending on the developer's machine: Windows/MacOS/Linux.**
- **Once VS Code is launched, the extension connects back to a C2 server.**

Piithon-linter: A malicious Python linter/formatter



VS Code Marketplace accepted the malicious extension again.

Backdooring the Extension via automatic malware execution

```
Merlin[listeners][c8ef72d7-1b0f-4a84-b10b-3391ea7f77ba]»
[+] 2025-04-14T21:22:30Z New authenticated Agent checkin for 29b7a59a-eee0-478e-8df7-c0e012b5f1cc at 2025-04-14T21:22:30Z
[-] 2025-04-14T21:23:02Z Results of job CjyQz0cxW for agent 29b7a59a-eee0-478e-8df7-c0e012b5f1cc at 2025-04-14T21:23:02Z
    Configuration data received for Agent 29b7a59a-eee0-478e-8df7-c0e012b5f1cc and updated. Issue the "info" command to view it.
Merlin[listeners][c8ef72d7-1b0f-4a84-b10b-3391ea7f77ba]»
Merlin[listeners][c8ef72d7-1b0f-4a84-b10b-3391ea7f77ba]»
```

Whenever anyone installs the malicious extension, or launches VS Code or Cursor AI after the malicious extension is installed, you will get access to the developer's machine through the Merlin framework.

Works on Windows, macOS, and Linux

Wait

Microsoft VS Code Marketplace runs analysis on the extension in a sandboxed environment.

How did the extension bypass the security checks?

Bypassing Microsoft Azure Sandbox Analysis

The extension blocked all execution from the Microsoft IP Ranges and networks.

No payloads are dropped when the extension is executed on Microsoft owned-environment.

```
sh-3.2$ whois 20.22.211.83
% IANA WHOIS server
% for more information on IANA, visit http://www.iana.org
% This query returned 1 object
```

```
refer:      whois.arin.net

inetnum:    20.0.0.0 - 20.255.255.255
organisation: Administered by ARIN
status:     LEGACY

whois:      whois.arin.net

changed:   1994-10
source:    IANA
```

```
# whois.arin.net

NetRange:    20.0.0.0 - 20.31.255.255
CIDR:        20.0.0.0/11
NetName:    MSFT
NetHandle:  NET-20-0-0-0-1
Parent:     NET20 (NET-20-0-0-0-0)
NetType:    Direct Allocation
OriginAS:
Organization: Microsoft Corporation (MSFT)
RegDate:   2017-10-18
Updated:   2021-12-14
Ref:       https://rdap.arin.net/registry/ip/20.0.0.0
```

```
OrgName:    Microsoft Corporation
OrgId:      MSFT
Address:    One Microsoft Way
City:       Redmond
StateProv:  WA
```



Wait

Microsoft VS Code Marketplace runs static analysis.

How did the extension bypass the static security analysis?

Piithon-linter: A malicious Python linter/formatter

```
if (await isAVRunning()) {  
    log('AV running, skipping');  
    return;  
}
```

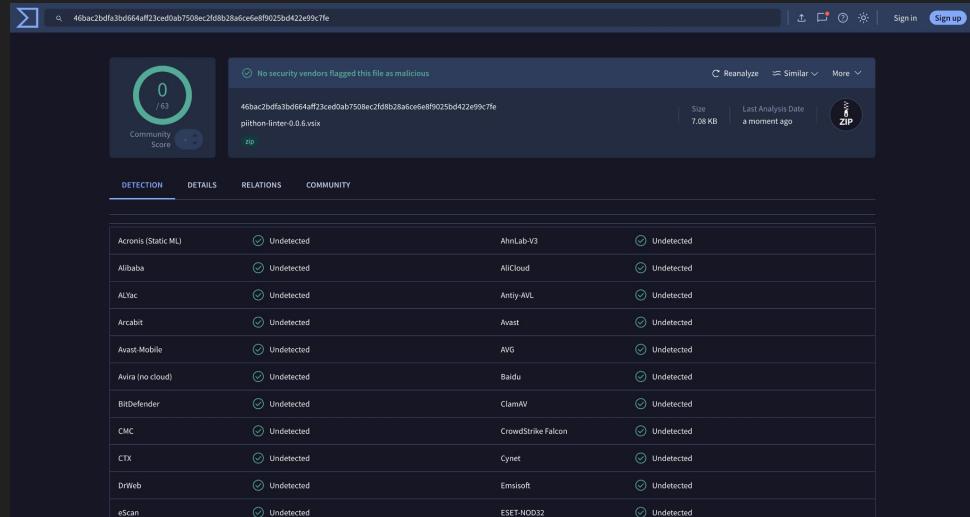
This line is copied from piithon-linter extension code.

The code contained plain-text indicators of malicious intent.
I'm not certain how the static analysis is currently functioning.

Malicious Extensions as an attack vector for Red Teamers

VS Code extensions provide a stealthy mechanism to stay under the radar.

You can also enable automatic updates, and allow VS Code Marketplace to distribute and deliver automatic malicious updates on developers' machines.



The screenshot shows a Fullhunt analysis interface for a file named '46bac2bdf3b0d64ff23ced0ab7508ec2f6b28a6ce6e8f9025bd422e99c7fe.zip'. The file is identified as 'python-linter-0.0.6.vsix'. The 'Community Score' is 0 / 63. A note says 'No security vendors flagged this file as malicious'. The 'Detections' table lists various security tools and their status:

Detection	Status	Detection	Status
Acronis (Static ML)	Undetected	AhnLab-V3	Undetected
Allisa	Undetected	AliCloud	Undetected
ALYsc	Undetected	Anti-AVL	Undetected
Arcabit	Undetected	Avast	Undetected
Avast-Mobile	Undetected	AVG	Undetected
Avira (no cloud)	Undetected	Baidu	Undetected
BitDefender	Undetected	ClamAV	Undetected
CMC	Undetected	CrowdStrike Falcon	Undetected
CTX	Undetected	Cynet	Undetected
DrWeb	Undetected	Emissisoft	Undetected
eScan	Undetected	ESET-NOD32	Undetected

LIVE DEMO

A realistic malicious extension with offensive capabilities

DEMO

Compromising Developers with Malicious Extensions

W Windsurf



THE BACKDOOR YOU DIDN'T SEE COMING

mazinahmed.net

DEMO

Compromising Developers with Malicious Extensions

Windsurf CURSOR



THE BACKDOOR YOU DIDN'T SEE COMING

mazinahmed.net

Responsible Disclosure

This is not going to be fixed soon

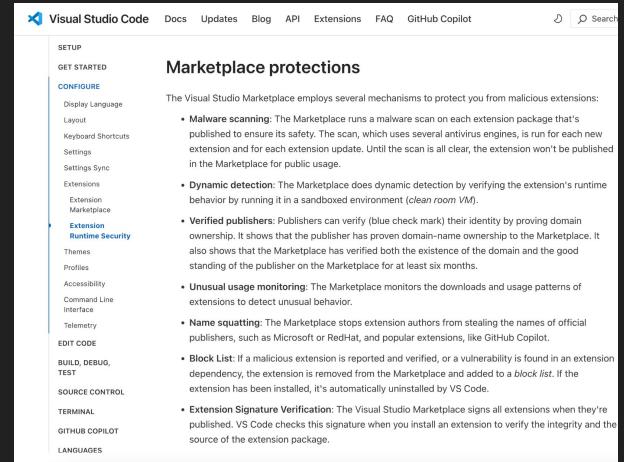
Responsible Disclosure

This is not going to be fixed soon.

Microsoft

- › ***After careful investigation, this case has been assessed as low severity and does not meet MSRC's bar for immediate servicing due to:***
- › ***There will be ways to bypass static analysis checks that are put in place to detect the problematic code. Therefore, it is the user's responsibility to ensure that they are not installing malicious extensions.***

I do not expect Microsoft to be resolving any of my findings within this research. All attack vectors mentioned here are available to adversaries to use and to bypass Microsoft VSCode Marketplace security controls.



The screenshot shows a sidebar menu for Visual Studio Code with categories like SETUP, GET STARTED, CONFIGURE, EXTENSION, and EDIT CODE. The EXTENSION category is expanded, showing sub-options like Extension Marketplace, Extension Runtime Security, Themes, Profiles, Accessibility, Command Line Interface, Telemetry, and GitHub Copilot. The main content area is titled "Marketplace protections" and discusses various security measures used by the Visual Studio Marketplace to protect users from malicious extensions.

Marketplace protections

The Visual Studio Marketplace employs several mechanisms to protect you from malicious extensions:

- **Malware scanning:** The Marketplace runs a malware scan on each extension package that's published to ensure its safety. The scan, which uses several antivirus engines, is run for each new extension and for each extension update. Until the scan is all clear, the extension won't be published in the Marketplace for public usage.
- **Dynamic detection:** The Marketplace does dynamic detection by verifying the extension's runtime behavior by running it in a sandboxed environment (clean room VM).
- **Verified publishers:** Publishers can verify (blue check mark) their identity by proving domain ownership. It shows that the publisher has proven domain-name ownership to the Marketplace. It also shows that the Marketplace has verified both the existence of the domain and the good standing of the publisher on the Marketplace for at least six months.
- **Unusual usage monitoring:** The Marketplace monitors the downloads and usage patterns of extensions to detect unusual behavior.
- **Name squatting:** The Marketplace stops extension authors from stealing the names of official publishers, such as Microsoft or RedHat, and popular extensions, like GitHub Copilot.
- **Block List:** If a malicious extension is reported and verified, or a vulnerability is found in an extension dependency, the extension is removed from the Marketplace and added to a *block list*. If the extension has been installed, it's automatically uninstalled by VS Code.
- **Extension Signature Verification:** The Visual Studio Marketplace signs all extensions when they're published. VS Code checks this signature when you install an extension to verify the integrity and the source of the extension package.

Responsible Disclosure

This is not going to be fixed soon.

OpenVSX

- › OpenVSX is an open-source and free project by the Eclipse Foundation (Non-profit organization).
- › AI-Powered IDEs rely on OpenVSX, and OpenVSX needs support to maintain the marketplace.

New key initiatives that the Eclipse Foundation is working on:

- Security scanning
- Improved authentication controls
- Provenance and verification

Responsible Disclosure

This is not going to be fixed soon.

Cursor AI

The Cursor Security team shared that they've rolled out new security features for users (built on top of Open VSX).

- › Do additional publisher verification
- › Changed the order Cursor presents extensions to further highlight legitimate ones
- › Integrated malware scanning on our side

I tried testing the malware scanning on Cursor AI and it did not detect the `piithon-linter` extension as malware.

Takeaways

Takeaway

**The next big supply chain compromise
could be from the editor we use every day**

Takeaway

Microsoft's sandbox analysis and antivirus checks can be bypassed with simple evasion techniques. OpenVSX, meanwhile, performs virtually no security screening at all.

Questions?

Mazin Ahmed

mazin@mazinahmed.net

Twitter: @mazen160

Linkedin: @infosecmazinahmed

Thank you!

Mazin Ahmed

mazin@mazinahmed.net

Twitter: @mazen160

Linkedin: @infosecmazinahmed