Mar 01 2024

NETMANAGEIT

Intelligence Report

New Malicious PyPI Packages used by Lazarus

Table of contents

Overview

Entities

Observables

External References

Overview

Description

JPCERT/CC confirmed that Lazarus has released malicious Python packages to PyPI, the official Python repository. The packages pycryptoenv, pycryptoconf, quasarlib, and swapmempool contain malware. The package names pycryptoenv and pycryptoconf target typos when installing legitimate packages. The malware is Comebacker, which decodes and executes a DLL sending HTTP requests to C2 servers. The DLL receives and runs executable files. The packages were downloaded 300 to 1200 times, showing Lazarus targets typos for infection.

Confidence

This value represents the confidence in the correctness of the data contained within this report.

100 / 100

Content

N/A

Indicator

Name
chaingrown.com
Pattern Type
stix
Pattern
[domain-name:value = 'chaingrown.com']
Name
blockchain-newtech.com
Pattern Type
stix
Pattern
[domain-name:value = 'blockchain-newtech.com']
Name
https://chaingrown.com/manage/manage.asp

Pattern Type
stix
Pattern
[url:value = 'https://chaingrown.com/manage/manage.asp']
Name
https://blog.phylum.io/crypto-themed-npm-packages-found-delivering-stealthy-malware/
Pattern Type
stix
Pattern
[url:value = 'https://blog.phylum.io/crypto-themed-npm-packages-found-delivering-stealthy-malware/']
Name
https://blockchain-newtech.com/download/download.asp
Pattern Type
stix
Pattern
[url:value = 'https://blockchain-newtech.com/download/download.asp']
Name

http://91.206.178.125/upload/upload.asp

Pattern Type

stix

Pattern

[url:value = 'http://91.206.178.125/upload/upload.asp']

Name

a8a5411f3696b276aee37eee0d9bed99774910a74342bbd638578a315b65e6a6

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'a8a5411f3696b276aee37eee0d9bed99774910a74342bbd638578a315b65e6a6']

Name

a4e4618b358c92e04fe6b7f94a114870c941be5e323735a2e5cd195138327f8f

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'a4e4618b358c92e04fe6b7f94a114870c941be5e323735a2e5cd195138327f8f']

Name

956d2ed558e3c6e447e3d4424d6b14e81f74b63762238e84069f9a7610aa2531

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '956d2ed558e3c6e447e3d4424d6b14e81f74b63762238e84069f9a7610aa2531']

Name

8fb6d8a5013bd3a36c605031e86fd1f6bb7c3fdba722e58ee2f4769a820b86b0

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '8fb6d8a5013bd3a36c605031e86fd1f6bb7c3fdba722e58ee2f4769a820b86b0']

Name

85c3a2b185f882abd2cc40df5a1a341962bc4616bc78a344768e4de1d5236ab7

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '85c3a2b185f882abd2cc40df5a1a341962bc4616bc78a344768e4de1d5236ab7']

Name

60c080a29f58cf861f5e7c7fc5e5bddc7e63dd1db0badc06729d91f65957e9ce

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '60c080a29f58cf861f5e7c7fc5e5bddc7e63dd1db0badc06729d91f65957e9ce']

Name

3ab6e6fc888e4df602eff1c5bc24f3e976215d1e4a58f963834e5b225a3821f5

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '3ab6e6fc888e4df602eff1c5bc24f3e976215d1e4a58f963834e5b225a3821f5']

Name

26437bc68133c2ca09bb56bc011dd1b713f8ee40a2acc2488b102dd037641c6e

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '26437bc68133c2ca09bb56bc011dd1b713f8ee40a2acc2488b102dd037641c6e']

Name

173e6bc33efc7a03da06bf5f8686a89bbed54b6fc8a4263035b7950ed3886179

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '173e6bc33efc7a03da06bf5f8686a89bbed54b6fc8a4263035b7950ed3886179']

Name

63fb47c3b4693409ebadf8a5179141af5cf45a46d1e98e5f763ca0d7d64fb17c

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '63fb47c3b4693409ebadf8a5179141af5cf45a46d1e98e5f763ca0d7d64fb17c']

Name

6bba8f488c23a0e0f753ac21cd83ddeac5c4d14b70d4426d7cdeebdf813a1094

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '6bba8f488c23a0e0f753ac21cd83ddeac5c4d14b70d4426d7cdeebdf813a1094']

Name

https://fasttet.com/user/agency.asp

Pattern Type

stix

Pattern

[url:value = 'https://fasttet.com/user/agency.asp']

Name

b4a04b450bb7cae5ea578e79ae9d0f203711c18c3f3a6de9900d2bdfaa4e7f67

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'b4a04b450bb7cae5ea578e79ae9d0f203711c18c3f3a6de9900d2bdfaa4e7f67']

Name

c56c94e21913b2df4be293001da84c3bb20badf823ccf5b6a396f5f49df5efff

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'c56c94e21913b2df4be293001da84c3bb20badf823ccf5b6a396f5f49df5efff']

Name

e05142f8375070d1ea25ed3a31404ca37b4e1ac88c26832682d8d2f9f4f6d0ae

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'e05142f8375070d1ea25ed3a31404ca37b4e1ac88c26832682d8d2f9f4f6d0ae']

Name

fasttet.com

Pattern Type

stix

Pattern

[domain-name:value = 'fasttet.com']

Name

91.206.178.125

Pattern Type

stix

Pattern

[ipv4-addr:value = '91.206.178.125']

Name

blog.phylum.io

Pattern Type

stix

Pattern

[hostname:value = 'blog.phylum.io']

Name

aec915753612bb003330ce7ffc67cfa9d7e3c12310f0ecfd0b7e50abf427989a

Description

Created by VirusTotal connector as the positive count was >= 10

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'aec915753612bb003330ce7ffc67cfa9d7e3c12310f0ecfd0b7e50abf427989a']

Name

01c5836655c6a4212676c78ec96c0ac6b778a411e61a2da1f545eba8f784e980

Description

Created by VirusTotal connector as the positive count was >= 10

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '01c5836655c6a4212676c78ec96c0ac6b778a411e61a2da1f545eba8f784e980']

Malware

Name
comebacker
Name
swapmempool
Name
quasarlib
Name
pycryptoconf
Name
pycryptoenv
Name
lazarus

Attack-Pattern

Name
T1060
ID
T1060
Name
T1573
ID
T1573
Description
Adversaries may employ a known encryption algorithm to conceal command and control traffic rather than relying on any inherent protections provided by a communication protocol. Despite the use of a secure algorithm, these implementations may be vulnerable to reverse engineering if secret keys are encoded and/or generated within malware samples/configuration files.
Name
T1064

ID

T1064

Description

This technique has been deprecated. Please use [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059) where appropriate. Adversaries may use scripts to aid in operations and perform multiple actions that would otherwise be manual. Scripting is useful for speeding up operational tasks and reducing the time required to gain access to critical resources. Some scripting languages may be used to bypass process monitoring mechanisms by directly interacting with the operating system at an API level instead of calling other programs. Common scripting languages for Windows include VBScript and [PowerShell](https://attack.mitre.org/techniques/T1086) but could also be in the form of command-line batch scripts. Scripts can be embedded inside Office documents as macros that can be set to execute when files used in [Spearphishing Attachment](https://attack.mitre.org/techniques/T1193) and other types of spearphishing are opened. Malicious embedded macros are an alternative means of execution than software exploitation through [Exploitation for Client Execution](https://attack.mitre.org/techniques/T1203), where adversaries will rely on macros being allowed or that the user will accept to activate them. Many popular offensive frameworks exist which use forms of scripting for security testers and adversaries alike. Metasploit (Citation: Metasploit_Ref), Veil (Citation: Veil_Ref), and PowerSploit (Citation: Powersploit) are three examples that are popular among penetration testers for exploit and post-compromise operations and include many features for evading defenses. Some adversaries are known to use PowerShell. (Citation: Alperovitch 2014)

Name

T1090

ID

T1090

Description

Adversaries may use a connection proxy to direct network traffic between systems or act as an intermediary for network communications to a command and control server to avoid

direct connections to their infrastructure. Many tools exist that enable traffic redirection through proxies or port redirection, including [HTRAN](https://attack.mitre.org/software/S0040), ZXProxy, and ZXPortMap. (Citation: Trend Micro APT Attack Tools) Adversaries use these types of proxies to manage command and control communications, reduce the number of simultaneous outbound network connections, provide resiliency in the face of connection loss, or to ride over existing trusted communications paths between victims to avoid suspicion. Adversaries may chain together multiple proxies to further disguise the source of malicious traffic. Adversaries can also take advantage of routing schemes in Content Delivery Networks (CDNs) to proxy command and control traffic.

Name

Obfuscated Files or Information

ID

T1027

Description

Adversaries may attempt to make an executable or file difficult to discover or analyze by encrypting, encoding, or otherwise obfuscating its contents on the system or in transit. This is common behavior that can be used across different platforms and the network to evade defenses. Payloads may be compressed, archived, or encrypted in order to avoid detection. These payloads may be used during Initial Access or later to mitigate detection. Sometimes a user's action may be required to open and [Deobfuscate/Decode Files or Information](https://attack.mitre.org/techniques/T1140) for [User Execution](https://attack.mitre.org/techniques/T1204). The user may also be required to input a password to open a password protected compressed/encrypted file that was provided by the adversary. (Citation: Volexity PowerDuke November 2016) Adversaries may also use compressed or archived scripts, such as JavaScript. Portions of files can also be encoded to hide the plain-text strings that would otherwise help defenders with discovery. (Citation: Linux/Cdorked.A We Live Security Analysis) Payloads may also be split into separate, seemingly benign files that only reveal malicious functionality when reassembled. (Citation: Carbon Black Obfuscation Sept 2016) Adversaries may also abuse [Command Obfuscation](https://attack.mitre.org/techniques/T1027/010) to obscure commands executed from payloads or directly via [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059). Environment variables, aliases, characters, and other platform/language specific semantics can be used to evade signature based detections and application control

mechanisms. (Citation: FireEye Obfuscation June 2017) (Citation: FireEye Revoke-Obfuscation July 2017)(Citation: PaloAlto EncodedCommand March 2017)

Name

Ingress Tool Transfer

ID

T1105

Description

Adversaries may transfer tools or other files from an external system into a compromised environment. Tools or files may be copied from an external adversary-controlled system to the victim network through the command and control channel or through alternate protocols such as [ftp](https://attack.mitre.org/software/S0095). Once present, adversaries may also transfer/spread tools between victim devices within a compromised environment (i.e. [Lateral Tool Transfer](https://attack.mitre.org/techniques/T1570)). On Windows, adversaries may use various utilities to download tools, such as `copy`, `finger`, [certutil](https://attack.mitre.org/software/S0160), and [PowerShell](https://attack.mitre.org/techniques/T1059/001) commands such as `IEX(New-Object Net.WebClient).downloadString()` and `Invoke-WebRequest`. On Linux and macOS systems, a variety of utilities also exist, such as `curl`, `scp`, `sftp`, `tftp`, `rsync`, `finger`, and `wget`. (Citation: t1105_lolbas) Adversaries may also abuse installers and package managers, such as `yum` or `winget`, to download tools to victim hosts. Files can also be transferred using various [Web Service](https://attack.mitre.org/techniques/T1102)s as well as native or otherwise present tools on the victim system.(Citation: PTSecurity Cobalt Dec 2016) In some cases, adversaries may be able to leverage services that sync between a web-based and an on-premises client, such as Dropbox or OneDrive, to transfer files onto victim systems. For example, by compromising a cloud account and logging into the service's web portal, an adversary may be able to trigger an automatic syncing process that transfers the file onto the victim's machine.(Citation: Dropbox Malware Sync)

Name

T1055

ID

T1055

Description

Adversaries may inject code into processes in order to evade process-based defenses as well as possibly elevate privileges. Process injection is a method of executing arbitrary code in the address space of a separate live process. Running code in the context of another process may allow access to the process's memory, system/network resources, and possibly elevated privileges. Execution via process injection may also evade detection from security products since the execution is masked under a legitimate process. There are many different ways to inject code into a process, many of which abuse legitimate functionalities. These implementations exist for every major OS but are typically platform specific. More sophisticated samples may perform multiple process injections to segment modules and further evade detection, utilizing named pipes or other inter-process communication (IPC) mechanisms as a communication channel.

Name

T1071

ID

T1071

Description

Adversaries may communicate using OSI application layer protocols to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server. Adversaries may utilize many different protocols, including those used for web browsing, transferring files, electronic mail, or DNS. For connections that occur internally within an enclave (such as those between a proxy or pivot node and other nodes), commonly used protocols are SMB, SSH, or RDP.

Intrusion-Set

Name
Lazarus

Domain-Name

Value
chaingrown.com
blockchain-newtech.com
fasttet.com

Url

Value
https://chaingrown.com/manage/manage.asp
https://blog.phylum.io/crypto-themed-npm-packages-found-delivering-stealthy-malware/
https://blockchain-newtech.com/download/download.asp
http://91.206.178.125/upload/upload.asp
https://fasttet.com/user/agency.asp

StixFile

Value
a8a5411f3696b276aee37eee0d9bed99774910a74342bbd638578a315b65e6a6
a4e4618b358c92e04fe6b7f94a114870c941be5e323735a2e5cd195138327f8f
956d2ed558e3c6e447e3d4424d6b14e81f74b63762238e84069f9a7610aa2531
8fb6d8a5013bd3a36c605031e86fd1f6bb7c3fdba722e58ee2f4769a820b86b0
85c3a2b185f882abd2cc40df5a1a341962bc4616bc78a344768e4de1d5236ab7
60c080a29f58cf861f5e7c7fc5e5bddc7e63dd1db0badc06729d91f65957e9ce
26437bc68133c2ca09bb56bc011dd1b713f8ee40a2acc2488b102dd037641c6e
3ab6e6fc888e4df602eff1c5bc24f3e976215d1e4a58f963834e5b225a3821f5
173e6bc33efc7a03da06bf5f8686a89bbed54b6fc8a4263035b7950ed3886179
63fb47c3b4693409ebadf8a5179141af5cf45a46d1e98e5f763ca0d7d64fb17c
6bba8f488c23a0e0f753ac21cd83ddeac5c4d14b70d4426d7cdeebdf813a1094
b4a04b450bb7cae5ea578e79ae9d0f203711c18c3f3a6de9900d2bdfaa4e7f67
c56c94e21913b2df4be293001da84c3bb20badf823ccf5b6a396f5f49df5efff

aec915753612bb003330ce7ffc67cfa9d7e3c12310f0ecfd0b7e50abf427989a

e05142f8375070d1ea25ed3a31404ca37b4e1ac88c26832682d8d2f9f4f6d0ae

01c5836655c6a4212676c78ec96c0ac6b778a411e61a2da1f545eba8f784e980

IPv4-Addr

Value
91.206.178.125

Hostname

Value
blog.phylum.io

Artifact

Value
703462497a8a85b00355ba7e572214fe84ea5151cd02adec6e76309fcaf06baf77e2846c3448ffb97ef8d8b0ad8b5edd2e434baad38eaa0f6855b04be461dcc7
1ecda3b9126c75f6019c6fa4bdfbe5f204617126a2f3349610bd81871c0f89e4aebdcdcebc68b396f5ed2cef62cd71e1645cc82cd8c95901f5322ff2507f0586

External References

- https://blogs.jpcert.or.jp/en/2024/02/lazarus_pypi.html
- https://otx.alienvault.com/pulse/65e0cb765723aa9cfaa6b362