Correlating Vidar Stealer Build IDs Based on Loader Tasks

insights.loaderinsight.agency/posts/vidar-build-id-correlation/

October 17, 2024

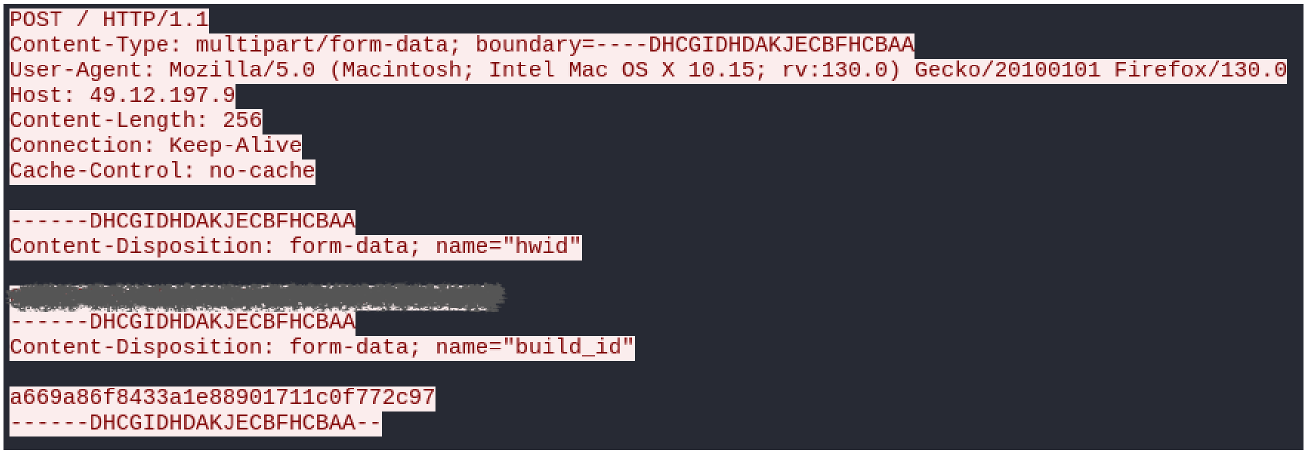
Posted Oct 17, 2024
By *LIA*
4 min read

Botnet Identifiers

When running a botnet, threat actors typically want to group bots together to keep track of campaigns or separate access in the administrative panel between users. This is the case in Malware-as-a-Service (MaaS) offerings where the malware author is providing threat actors access to existing infrastructure to manage bots and access information specific to their botnets. When a bot contacts its configured command and control (C2) server needs to notify the C2 server to which group/botnet it belongs to.

The type of identifier used to differentiate between the various botnets varies between malware families where SmokeLoader uses a plain string such as `pub1` or `pub2` while other families like Bokbot uses a binary identifier (referred to as `project ID`) which is mapped to a readable string in the backend for the threat actor's convenience. That said, there are families who does not have any identifiers and thus operate as one big botnet.

Vidar stealer uses a 32 byte hexadecimal string which is referred to internally as `build_id`, shown in the below screeshot.



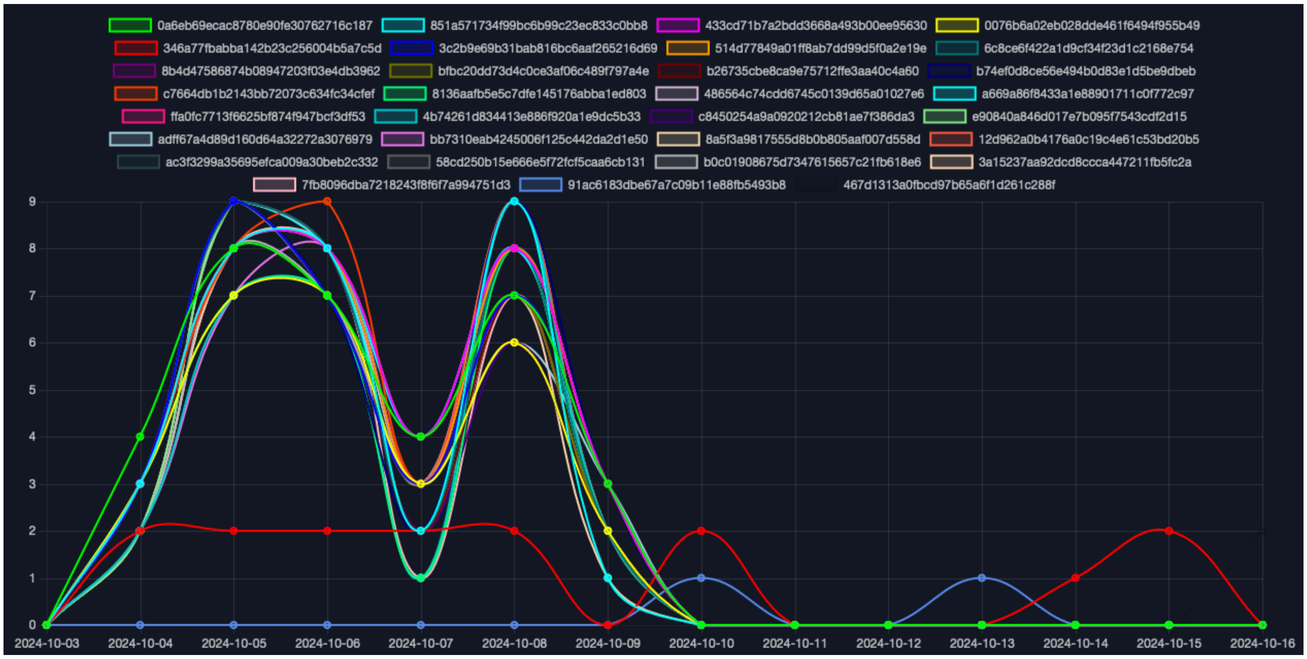
```
POST / HTTP/1.1
Content-Type: multipart/form-data; boundary=----DHCGIDHDAKJECBFHCBAA
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:130.0) Gecko/20100101 Firefox/130.0
Host: 49.12.197.9
Content-Length: 256
Connection: Keep-Alive
Cache-Control: no-cache

------DHCGIDHDAKJECBFHCBAA
Content-Disposition: form-data; name="hwid"

[redacted]
------DHCGIDHDAKJECBFHCBAA
Content-Disposition: form-data; name="build_id"

a669a86f8433a1e88901711c0f772c97
------DHCGIDHDAKJECBFHCBAA--
```

The name hints that a when a threat actor receives a new build from the malware author it comes with a new unique identifier that is mapped to the threat actor's profile in the Vidar backend. This presents a problem from a researcher's perspective as two samples with two different build IDs cannot be grouped together without having access to the Vidar backend. However, with access to historical tasking associated with each `build_id`, it is possible to correlate multiple build IDs over time and build a larger picture of the actor's campaign.

Vidar Build IDs & Lumma Stealer Tasks

During our tracking of Vidar we have gathered data on not only tasks threat actors distribute but also their build IDs making it possible to correlate build IDs with their respective tasks and subsequently tie a group of build IDs to the same threat actor.

At the beginning of October we saw an increased amount of activity with a larger number of build IDs being used. The below graph shows the build IDs and the unique number of payloads that were collected as a result of the tasks that were distributed by threat actors.



Out of the 31 build IDs in the graph, 28 of them demonstrate a pattern where a large amount of unique payloads are distributed. A closer look at the tasks associated with these build IDs showes that only two types of tasks are being distributed, as shown in the below table.

Task URL	Unique Payloads
http://nsdm.cumpar-auto-orice-tip.ro/ldms/a43486128347.exe	63 payloads
http://jask.powerforxes.shop/ldms/a43486128347.exe	68 payloads

The two tasks have resulted in a total of 131 unique payloads, identified by UnpacMe as Lumma Stealer. In addition, the same two tasks have been observed distributed by StealC with botnet identifiers `default` and `us_test`, potentially providing a pivot point in tracking the threat actor in question. The amount of payloads over the course of the campaign suggests that the threat actor likely had automation in place to update the payload as an attempt to evade detection. An example of this type of automation was documented by RussianPanda in connection with DolphinLoader.

Notes From the Trenches

The above insights are made possible by LIA's continuous intelligence collection. We track malware loaders in real time, and over time, producing timely IOCs, and enabling big-picture analysis. All the shared data (and more) is available in the LIA platform for researchers to work with. It is possible to search for botnet identifiers such as Vidar build IDs in LIA which can provide analysts and researchers a glimpse into historical tasks for that have been distributed and subsequently be able to provide more context to an ongoing investigation.

Indicators of Compromise

Lumma Stealer Payloads (SHA256)

SHA256
6296f2f7c26fefaed8b960f94ccc3c85f40ebe0275b6b5b5f9baccd71bf0c7c6
3ca7a9fb631de6c3d108f2f7c016fdd447ff1df32c0d65fd3e3d8c77d0709c33
80aa04dff22601d8ee92fca945fa93d80993d02c6fe6da339d9459994159dbcb
3642e58272247143e3d60ae51738f4bd89fe52e41237f9ceaff0642d482d85e3
f796bcbd4b09d49180d27bfb02cbe05071e297663ffcf14c44d55fd09a6adb8f
9fa637487aed3d08b3fb1ebc0fceefa22d7df778464893efcb07390f172c337f
87818d5d5b90e230b668fffeb42f1e4c035b16ac80c611698e6b3351d6c1e27a
ef2048a669f1bf5ddf52eaf0f3950f5fa6c8eabc185eb8f6fa1a4af7731e6442
cc786b72fd3b6f4165a962538cf43992d20620b6794dc47c3af58988ff8dd417
a00ac6b0e80396c240359f30f6d74a4c30e76ab64165dd9d94a39d72946c71aa
c5f9b73bc19113661646077adade8074f01ed27c67816514a26fbe9edbc30743
f123a7962728b2a9c22f21ef131de3bf84b4a3ce4c08c0de90a77c3027bbb7b8
3eebc4cba36e1bbafd17ddaf933e3620489f7eba83f5d56b64d538b374c594f5

SHA256
523e991d04e1487ba9dce934700d54c16e53c1e4025d3493705d704f576fcd6e
5215c0dd69ea30cbf647a75099d2032b13dfa6bd93021ce5a1a239c016a90a31
d29b4b63c9bb58a662c28b69013aa91ab5923a116e8fc21c8f5bbc45672fc780
7034f6cd99479264e2982b174aa53d60a331be298bfae46fc9c0ae689bdb18c7
b105ed7d2bd9ab74ba33c6c9dbdd0aa86134ab9c50a70338bfdec909e4bc5165
8f83a1dc767ac821c811cf71d05fd1ac3f499fdba2e74699f56f5f221bbb2a30
e712c288e4281759f94c5ccb1ea3aa637de816b046d633e6b072345a0c749842
9d5a09f1e7085458979b815665c0065b267923a04a359bda91cebdd0e3ba7758
edd3e06d019a58ce34e72b10f8e364d62fe3df26574c6b4d11065d07684c9cb6
3027102cf7924a85538761ee07550d64652d0dd1b28d7e59b6c9f46b2aea2d05
3d6a2eff7723098111c040dd15ae53f43335dc79f31fcfb7308864cb0b1b522a
fff469313f3f33af2ce8be9fc434cb0f8102d7927d28f861cc8edd2c4d7738b2
2b3fb4358725421127db0e62741eaa33741a98f95d364b5d4914f785718fe665
d0c8e84e22df417a0a20578f1adfa1f12720acb9880c599355be82bb0c6659a8
5415cbc13a8030b546266d4a22cd68731100479157e86982f98734be058493ec
53e5a4e6091e7a99011b3e394b95684ac0275d22336d3852036a60339dfd0143
970a3479516eff2a2c2e016150731a1bb5d1324267e38cd9a7852330840adc6a
7b467a82e1138ade052abc2151d1ebfd6d5dae99fc76f2a7a9f8c9d95d66eae2
ba78ce8695583f77a425e268f2ee214094cc24513129e5b41bd270ffa6286406
5403c33d339c9dad1444ad85d86fe8acfc22c8e95c39ca269606210431aa365d
96e22dd2341c0b8c1967aeeaeae7157485cb987c0f82e166f037c35059eeea65
4e33df055827829444910db4e73d33fa9315cc24e0be89467abf8cb2c1fcfb87
2f4fef756205a42d7c431631fcb2f2cd8b336c504b5360b39d81cc52a3a625ec
302de22282260881f8ff0ecfd710943bb6f80418fe63e1607522f84b83a5a40d
7e822f2c628b3919b08d9971d8f32bb03e1944df06ae3e0306975b1609e6f35e

SHA256
b64ac8b37344d0d295fb522c83247aad117a5d963c0bc2d3867f8744d8d34f0a
e7e50cb72d213404239b50c90536c4f12dae0c7666827fa4d15de6b0c0cd8666
69f2ee8f7058c5c73da7641d06fbfe9e1e5c5c27da7016fa8b9a85991a3595df
2682d605469d831b4c28f938d4eb94c67711a82b0d094340dcb31186c69e98a3
dd2f9b5da37a98db666f9c3740127555179c1d31b2d5865a0e2890c5c1fbf305
d8bd331094a065c93ec5c1cc3fa34a1eba2616f990ffd93edf3272d388fe06f4
69be73d4bca2e52f9f11964fe7346e549c346ac197a25de252f60c7103e5dfe2
a898c26db527f30fa45c58c08f71ca8a0c8458a665cc4046be36ca0bb3ac80e3
826f0ef12cdaffa5a8c1f3cdd94fe9edce83f6c7bb54599524ff71f6f6321a5b
82f0028433d8c523057043a117f95170b143440b609a534547a53933e656725c
503ae54f5e87835c5e96838c63e6bbdfcfb9cad9246375a283074f68af8dcdb4
3026b086b3793824ca9e3090d06c8d6e0222372624b71d57bcefc40c94e1a0b1
c457134d21ff78a008fbc58dbde5ff6d3ebf55564d86541328eb76eb83304bea
5938cd354f86ee5c58349a995dac3bcfd593bf5920fe46d080f13fd93a65beb2
b262c671d9ba5ffcd32c34d13f33b87cf1039640cc813e318a35f0f2a3db2161
0e60dabbdc133ff872b0c690604566a55f8351178f31d7012fccf256c5fe27e2
c9b8299487ceb8d19fa9fed139c90b0f9f0bc0a3763954e24c2e6d62aaf22960
82afd27dea78a6278055c1e46506518b28483ae5ffb50346b2ea85003562dc98
f6bba2d93711805269a9ae75ef72f380a6e7de9229cb891f2a6e47dc17755c00
0ba6befd6c519b35700a6a4ff5d695c3eeaaebe9cbec9b89ffb9d9da1633a5ae
3cd8799b4301cb1a6dc747f054d0c769e76798a128b17c0b7163c7170da71a1f
5fd22fb35f241e7642432f6ec3a479437ce90d6d3f2a50b52431b607208b685d
6f3faad7580926cd2112ee03aa26376fa04175e6c5e2ef00390fd5c1ca45a741
9c04149a55f07d041f4abbfd4a1ed84cea5e2cbb47d87399f33c6e3d7f76f228
9037df6561a0aedab6049df8dade28ed2eb0b830d0eb6061d8a7e633c1388303

SHA256
0d46ca7d7534ef2800b153bd16979af898ef74f1f64854c1a3f9fbc8d14a0d57
86db2c9e3c8e0a43dd31a32e76c0527a1658b9f393614edf3827803c38b54b90
961e8be766b3f4429ef7011c387044c07bb0c229a55bba7b2ad0492134349ca6
6904852a1ea37a509612e8ea2382ad12e35bcd8b69f780110ec5c9e78eea73ef
7d2cb4015ae0d46271a8f05a963cafe33badafe52869c330d36621c5e65bcf9c
6c7818a65f46711fbc89cd7b548829e98be247fab8b2c4766c85b64bc632e797
4167b003ef7a7a9120b73685d3b04acc8055e0ad8728a103a8a7fd08bdc3c142
ead35a9130d317fb0a615969e9136596ef244092bffdd92ae05ef46e1bf63ce9
1445f8dc16bf7f0e1c7b3d16bee14ef83e6170ab00a2381d509051c64617fbfd
53b4e4b1016a463835a05163aceebbaceb25f0edd24b7dfc8c02be3596a65e87
8f7eac152de0b15ebf4de4103f9b4f9a05547c7389fc17f51a5ada2acb556f4b
987948d987c40aa32a075898caef9e860d056a3d8a2404fbc95c3ebc0d3290a4
276316da845aef57b659e91e593641e6a6b267e2bb7bae0b4afec671809322a4
cb65f205570f447f582985450264005a1ea6f87f22f98fe90be13537267724ee
7781fd500447528bfa5f7064c773b07caaf1d13ba734aca1d10dc95d401c07fe
5639a11bef6a2b25409cbdb97cd1f36601a334a1aa80b7009043cd88af0913a4
ab39cac07069e4508158087ad82cee09b55d9bebe56ed1b9670863d90f1e1edf
acc8d2507908025af3a013002f9d97b935ea22da07a69903109ce3fd93e78b6f
b1b6ccf4681d86a4a18cdab8318739887fb36eb7cfa0f8f916d011c8685ea944
93bd0a8b49c37ae06b2f371f4fd1be25228033121411ee4b19bdf932396e94ec
f5fc12afcc52c3dbee5cbb21b2d2181e6ab3cb731f8f057f8a4772ed3bc52c9b
7cae5be06227549a4c05c88e05313954539d137e55ae5b0bb73abc5981c21641
ca03b8d8929a2c6a1e94663b3b45a1d46b6e5002f13858c8dc05a83d5b11c607
14dd3ea029df5fddf05dbc2e9d920fb1bbfc2ebad1c66c5099ffd68874dd431e
84f9830b538a6dc944b35532e5d326a246b0ba8861ec6e19bea213dd71372f2b

SHA256
f44d05e3b61f4690782ce7e72ff24209abeea31013e50dd8233ef9c2e5f3a6a4
b9de370b333f88f3a2e5f7017510199bcf2799d465a240b4c3ba015499cf129e
d485784fa9dc14aa2885f703a400b89256caf36fe85a9a5f2c30496bc204af1c
6a1da56f0f9c29264a5537e81d3f90a6785c5ebd36d3428094d66e708bf80c7e
7fa92a341696337fbcf5b0ac834b143373a4aa59afbfa838c0d8e8368568c904
1fc5ef14137ea4754c446e675127bc1bb5f903e527b74b357f574bf47150a75c
567383e7b9842a00300af751e9a92a88c8ec1347d5795a0ffb9cc7c488bff51b
28d219abcb6751f4c7c317d5ec4323244a09d0ec2d2750262afb360408385bce
9765a5d34338ac761f08823314d7f9d54fe72e5c98440ce80780c9b49fd5026a
cdbbbbf877458ce603599fb9f9f445eac37a00dc67d065f141485a8d9f1df644
92a98663c73a42b131e32998dc5e9998f24e5f03d712857a33478bbf96d4c761
ffeabfc0d6b33647e95b723947a3fa8e174a91ebc370f1a7528c74ed4d3193ed
04d92121f16ca15e94cc3d693f03e114dea67c84905904c716d5e6d877827cbd
d847f1834d945e4ac199083136efa6baca021e30c995a78cc338e6145d7ff121
e8c8996a0fa7dbafa5864502b953c4c0b60dcc189acdcd5d019ae30d411c9375
7137aa46422d4df9531daa8beb0636bb86ac15aacfad74f73ee35155373d2a49
596667fd7f685701e6b4d0052b0996b9aaff795048cc1f3df2f8afc50a1e9f86
02463503eec09497a5d56c2573d77f5a5bd0d215dae8d95a60eea4d965b9ee32
ab3b23e3154fcf362b142153c5f334b5a565ff85cd59d95b1d498d50d740ed8e
0ceb0298250220b81dcb1e1059de1d5a6c19575912eff535c3524f8d358f3f98
ecaea1b071baa4e4267388c372de133fa6ac3b515f052e2fdf058a6c4a64c589
fa7883fae6e8a981551d097bd4f6c8cf0d9bba25e9189b422baf6531a5e970be
3ed0b5c68e92806325c5fe1813b96938d7666f1d2ad86e28b6a0d1beb0c57ed3
74d77b22d6fcefb7cb013835fdc25bb67b20f5eef2b2b58df91bf1b3342bcf1e
aceaeb8e3cfce22c78b502938310ead45cf0ce225bc5163d280ef71a27fed91f

SHA256
3133ec7f157cc16c4096df439faceb6995e1e0b5ede3668eadb8cfc24fed98ce
dee89b739072d5bf4c3389e562fe1c8fe63d33ddc8990517f7e8ea5a3c852522
d8e5f380c483313146d4998747db659e530198a9ddfabc47cf87e0562201dd93
917939ceb07cacd6a5960d799d9c4de954a07b93c135b54c39e1fc4a798d1566
d6303a71ac1f6fd6ddeab804764cbcd46fd53f8ec8f258238dd60c0c1f2fca4c
fd8c183d915e1412e491848342473fd26cb6e9543a1679acd15f85bb7f3dafa6
28ac2684af5fc6519b3af6424f5603e0a398ab4422f6b5f05e5d786fc596902e
7ca7eef2fa3e58ca002663723d4a3e4fa7a39b9ba6a96314315cdd00aa20d76a
ab7152a55aaf8d7ebd93a3261894eff791712f8dc205b653ce498a2f6ec20116
34638b10d3dd1387d419b8a81ed487fb0f052178f9ba231d9346109b1526bcd6
433747e0bb9824df3ecd109b8a595ba156895af40d83383149375e798cc81b85
58e87543c2167698b1bd4630722731971dcfb203e1c4172b576edeed29682c34
c0f89aed08873352dcd91f3d98f7284902a9fe0ea884ab1e66132f5d9c8837a3
f5378f0beb792d72f752ad36b3e9432f3303cd849fade3f68c811d761d23756a
a70dd29582751362c3b63a175a44db988ffb0aff4898705e3cdf5112e6e070b6
1a068d24b243557ed75d71e6d5866ca1a1d6a63fadcd672ddec9e836b83444d2
c07557cc8cf976f2f0d093d193fa1901bd0010261555a7f3624dbead45ad8f9a
6ad1c9a514dd9a2e2213d52f6c943451c9eb8f2a074913fa98c68d644fd3466b

Vidar Build IDs

Build ID
0a6eb69ecac8780e90fe30762716c187
851a571734f99bc6b99c23ec833c0bb8
433cd71b7a2bdd3668a493b00ee95630
0076b6a02eb028dde461f6494f955b49
3c2b9e69b31bab816bc6aaf265216d69

Build ID
514d77849a01ff8ab7dd99d5f0a2e19e
6c8ce6f422a1d9cf34f23d1c2168e754
8b4d47586874b08947203f03e4db3962
bfbc20dd73d4c0ce3af06c489f797a4e
b26735cbe8ca9e75712ffe3aa40c4a60
b74ef0d8ce56e494b0d83e1d5be9dbeb
c7664db1b2143bb72073c634fc34cfef
8136aafb5e5c7dfe145176abba1ed803
486564c74cdd6745c0139d65a01027e6
a669a86f8433a1e88901711c0f772c97
ffa0fc7713f6625bf874f947bcf3df53
4b74261d834413e886f920a1e9dc5b33
c8450254a9a0920212cb81ae7f386da3
e90840a846d017e7b095f7543cdf2d15
adff67a4d89d160d64a32272a3076979
bb7310eab4245006f125c442da2d1e50
8a5f3a9817555d8b0b805aaf007d558d
12d962a0b4176a0c19c4e61c53bd20b5
ac3f3299a35695efca009a30beb2c332
58cd250b15e666e5f72fcf5caa6cb131
b0c01908675d7347615657c21fb618e6
3a15237aa92dcd8ccca447211fb5fc2a
7fb8096dba7218243f8f6f7a994751d3