How to improve and speed-up DFIR with hashlookup

Indexing all the published software

CIRCL
Computer Incident
Response Center
Luxembourg

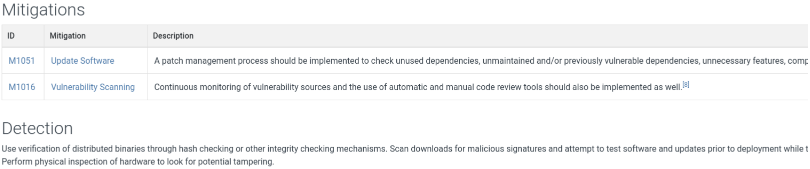
Alexandre Dulaunoy
TLP:WHITE

info@circl.lu

Unlock Your Brain, Harden Your System

ATT&CK Technique: Supply Chain Compromise (T1195)

- *Adversaries may manipulate products or product delivery mechanisms prior to receipt by a final consumer for the purpose of data or system compromise.*
- **Use verification of distributed binaries through hash checking** but is this easy? where to find those hashes?



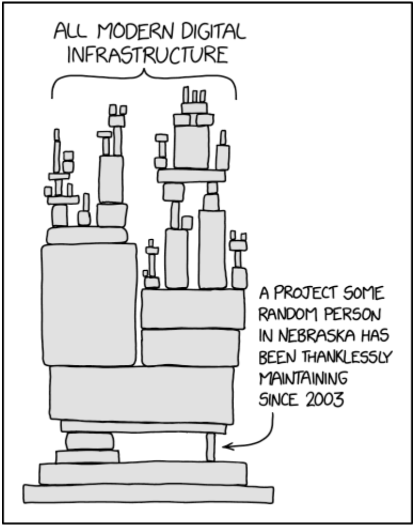
Mitigations

ID	Mitigation	Description
M1051	Update Software	A patch management process should be implemented to check unused dependencies, unmaintained and/or previously vulnerable dependencies, unnecessary features, comp
M1016	Vulnerability Scanning	Continuous monitoring of vulnerability sources and the use of automatic and manual code review tools should also be implemented as well.[8]

Detection

Use verification of distributed binaries through hash checking or other integrity checking mechanisms. Scan downloads for malicious signatures and attempt to test software and updates prior to deployment while t
Perform physical inspection of hardware to look for potential tampering.

Do you know about this little binary used everywhere?


ALL MODERN DIGITAL INFRASTRUCTURE
A PROJECT SOME RANDOM PERSON IN NEBRASKA HAS BEEN THANKLESSLY MAINTAINING SINCE 2003

Starting digital forensic investigation on a recent acquisition

- A single disk acquisition of a desktop or server operating system can contain at minima 150K files,
- Large portion of directories and files are not analysed due to a **lack of time**,
- Finding legitimate versus attacker-installed files can be difficult if the timeline is incorrect,
- Many legacy tools are used by attackers and mixed with custom binaries.

Known file filters - DFIR issues

- **State of current NIST NSRL**[1] databases and other known file filters (KFF),
- too few Operating Systems / Software available (e.g. OSX?, Linux distributions),
- nsrllookup.com / nsrlsrv use their own protocol, no ReST API
- nsrlsrv[2] only support MD5,
- many **sources are difficult to use** (e.g. NSRL ISOs), **ill-maintained** or **outdated** or **expensive**,
- MISP integration (malicious hashes versus known hashes).

[1] `https://www.nist.gov/itl/ssd/software-quality-group/national-software-reference-library-nsrl`

[2] `https://rjhansen.github.io/nsrlsvr/`

Indexing all published software?

- **Regular updates of Linux distribution** including security updates on multiple architectures,
- 800+ software releases per hour on GitHub,
- Bundling of software in **snap** images, **flatpak**, **AppImage**, etc.
- **Continuous release** of security updates,
- Microsoft Windows and Apple custom software distribution schemes.

Known file filters - improvements

- The need of a **public, open and easy** to use API for all sources (NSRL is not alone),
- a **global public instance of all known sources**,
- a common ReST API normalizes the access to several datasources,
- available for MD5, and SHA1 (and more),
- that includes fuzzy hashes,
- additional datapoints available through the **intersection of datasources**.

CIRCL hashlookup public service

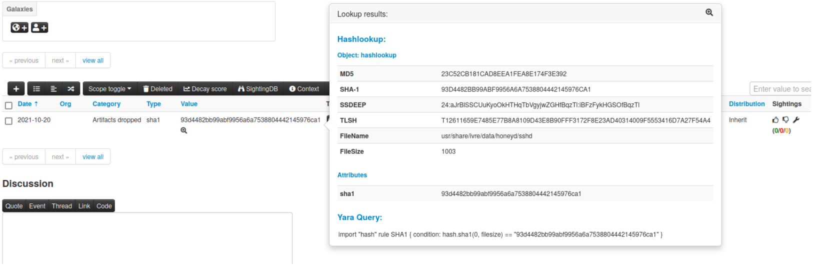
- https://hashlookup.circl.lu/[3] - **OpenAPI** Swagger[4]
- NIST NSRL - **all RDS hash sets** including current, modern, android, iOS and legacy,
- Ubuntu packages distribution,
- CentOS core OS distribution,
- Fedora project EPEL repository,
- CDNjs repository,
- Kali linux packages distribution, OpenSUSE distribution and **more**,
- **If you find it in a lot of trusted places, you may find that it's reasonable to trust it**.

[3] https://hashlookup.circl.lu/

[4] https://hashlookup.circl.lu/swagger.json

hashlookup MISP module

- A hover and expansion module[5] to quickly check if a hash is part of the known files of hashlookup:



□	Date ⇡	Org	Category	Type	Value	T	Distribution	Sightings
□	2021-10-20		Artifacts dropped	sha1	93d4482bb99abf9956a6a7538804442145976ca1		Inherit	(0/0/0)

Lookup results:

Hashlookup:

Object: hashlookup

MD5	23C52CB181CAD8EEA1FEA8E174F3E392
SHA-1	93D4482BB99ABF9956A6A7538804442145976CA1
SSDEEP	24:aJrBlSSCUuKyoOkHTHqTbVgyjwZGHfBqzTl:iBFzFykHGSOfBqzTl
TLSH	T12611659E7485E77B8A8109D43E8B90FFF3172F8E23AD40314009F5553416D7A27F54A4
FileName	usr/share/ivre/data/honeyd/sshd
FileSize	1003

Attributes

sha1	93d4482bb99abf9956a6a7538804442145976ca1

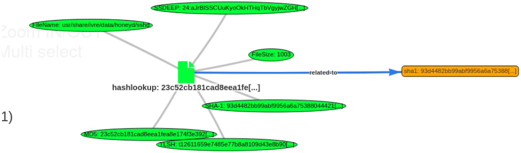
Yara Query:

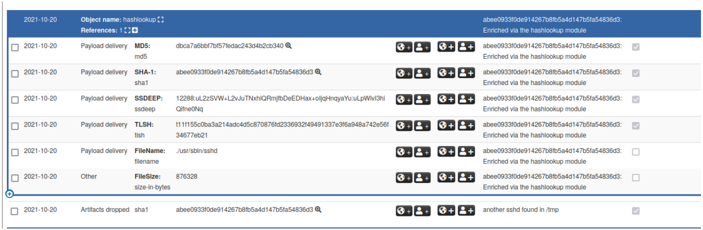
```
import "hash" rule SHA1 { condition: hash.sha1(0, filesize) == "93d4482bb99abf9956a6a7538804442145976ca1" }
```

[5] https://misp.github.io/misp-modules/expansion/#hashlookup

hashlookup MISP module - import

1)

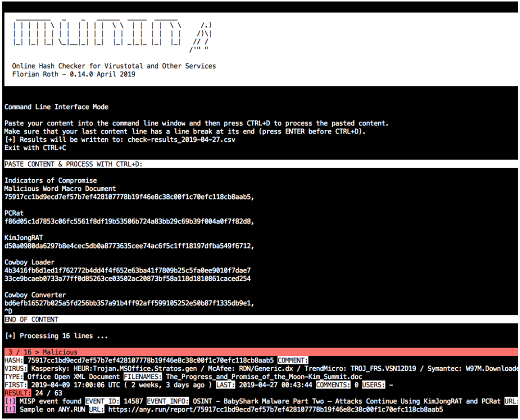

Zoom IN/OUT
Multi select
SSDEEP: 24:aJrBISSCUuKyoOkHTHqTbVgyjwZGH[...]
FileName: usr/share/ivre/data/honeyd/sshd
FileSize: 1003
related-to
sha1: 93d4482bb99abf9956a6a75388[...]
hashlookup: 23c52cb181cad8eea1fe[...]
SHA-1: 93d4482bb99abf9956a6a75388044421[...]
MD5: 23c52cb181cad8eea1fea8e174f3e392[...]
TLSH: t12611659e7485e77b8a8109d43e8b90[...]



	2021-10-20	**Object name:** hashlookup **References:** 1			abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	
□	2021-10-20	Payload delivery	**MD5:** md5	dbca7a6bbf7bf57fedac243d4b2cb340	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	☑
□	2021-10-20	Payload delivery	**SHA-1:** sha1	abee0933f0de914267b8fb5a4d147b5fa54836d3	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	☑
□	2021-10-20	Payload delivery	**SSDEEP:** ssdeep	12288:uL2zSVW+L2vJuTNxhiQRmjfbDeEDHax+oljqHnqyaYu:uLpWlvI3hiQifne0Nq	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	☑
□	2021-10-20	Payload delivery	**TLSH:** tlsh	t11f155c0ba3a214adc4d5c870876fd2336932f49491337e3f6a948a742e56f34677eb21	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	☑
□	2021-10-20	Payload delivery	**FileName:** filename	./usr/sbin/sshd	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	□
□	2021-10-20	Other	**FileSize:** size-in-bytes	876328	abee0933f0de914267b8fb5a4d147b5fa54836d3: Enriched via the hashlookup module	□
□	2021-10-20	Artifacts dropped	sha1	abee0933f0de914267b8fb5a4d147b5fa54836d3	another sshd found in /tmp	☑

Munin - Online hash checker

- Munin[6] is a online hash checker utility that retrieves valuable information from various online sources including hashlookup.



```
 _________   _    _   ______  _____  ______
| | | | | \ | |  | | | |  \ \  | |  | |  \ \     /.)
| | | | | | | |  | | | |  | |  | |  | |  | |    /)\|
|_| |_| |_| \_|__|_| |_|  |_| _|_|_ |_|  |_|   // /
                                              /'" "

Online Hash Checker for Virustotal and Other Services
Florian Roth - 0.14.0 April 2019


Command Line Interface Mode

Paste your content into the command line window and then press CTRL+D to process the pasted content.
Make sure that your last content line has a line break at its end (press ENTER before CTRL+D).
[+] Results will be written to: check-results_2019-04-27.csv
Exit with CTRL+C

PASTE CONTENT & PROCESS WITH CTRL+D:

Indicators of Compromise
Malicious Word Macro Document
75917cc1bd9ecd7ef57b7ef428107778b19f46e8c38c00f1c70efc118cb8aab5,

PCRat
f86d05c1d7853c06fc5561f8df19b53506b724a83bb29c69b39f004a0f7f82d8,

KimJongRAT
d50a0980da6297b8e4cec5db0a8773635cee74ac6f5c1ff18197dfba549f6712,

Cowboy Loader
4b3416fb6d1ed1f762772b4dd4f4f652e63ba41f7809b25c5fa0ee9010f7dae7
33ce9bcaeb0733a77ff0d85263ce03502ac20873bf58a118d1810861caced254

Cowboy Converter
bd6efb16527b025a5fd256bb357a91b4ff92aff599105252e50b87f1335db9e1,
^D
END OF CONTENT

[+] Processing 16 lines ...

 3 / 16 > Malicious
HASH: 75917cc1bd9ecd7ef57b7ef428107778b19f46e8c38c00f1c70efc118cb8aab5 COMMENT:
VIRUS: Kaspersky: HEUR:Trojan.MSOffice.Stratos.gen / McAfee: RDN/Generic.dx / TrendMicro: TROJ_FRS.VSN12D19 / Symantec: W97M.Downloade
TYPE: Office Open XML Document FILENAMES: The_Progress_and_Promise_of_the_Moon-Kim_Summit.doc
FIRST: 2019-04-09 17:00:06 UTC ( 2 weeks, 3 days ago ) LAST: 2019-04-27 00:43:44 COMMENTS: 0 USERS: -
RESULT: 24 / 63
[!] MISP event found EVENT_ID: 14587 EVENT_INFO: OSINT - BabyShark Malware Part Two - Attacks Continue Using KimJongRAT and PCRat URL:
[!] Sample on ANY.RUN URL: https://any.run/report/75917cc1bd9ecd7ef57b7ef428107778b19f46e8c38c00f1c70efc118cb8aab5
```

[6] https://github.com/Neo23x0/munin

Other services or tools using hashlookup API/db

- metalookup.com - Find published software by hashes,
- The Hive Project - Cortex analyser,
- **hashlookup-forensic-analyser**[7]: a script to analyse a forensic target,
- Add your tool? CIRCL hashlookup API is freely accessible.

[7] https://github.com/hashlookup/hashlookup-forensic-analyser

hashlookup references

- **hashlookup org on github**[8]:
  - **hashlookup-format**[9]: Common output format for hashlookup
- **Public API**: `https://circl.lu/services/hashlookup/`
- **hashlookup-format draft IETF** [10]
- Contact: info@circl.lu

[8] `https://github.com/hashlookup`

[9] `https://github.com/hashlookup/hashlookup-format`

[10] `datatracker.ietf.org/doc/draft-dulaunoy-hashlookup-format/`