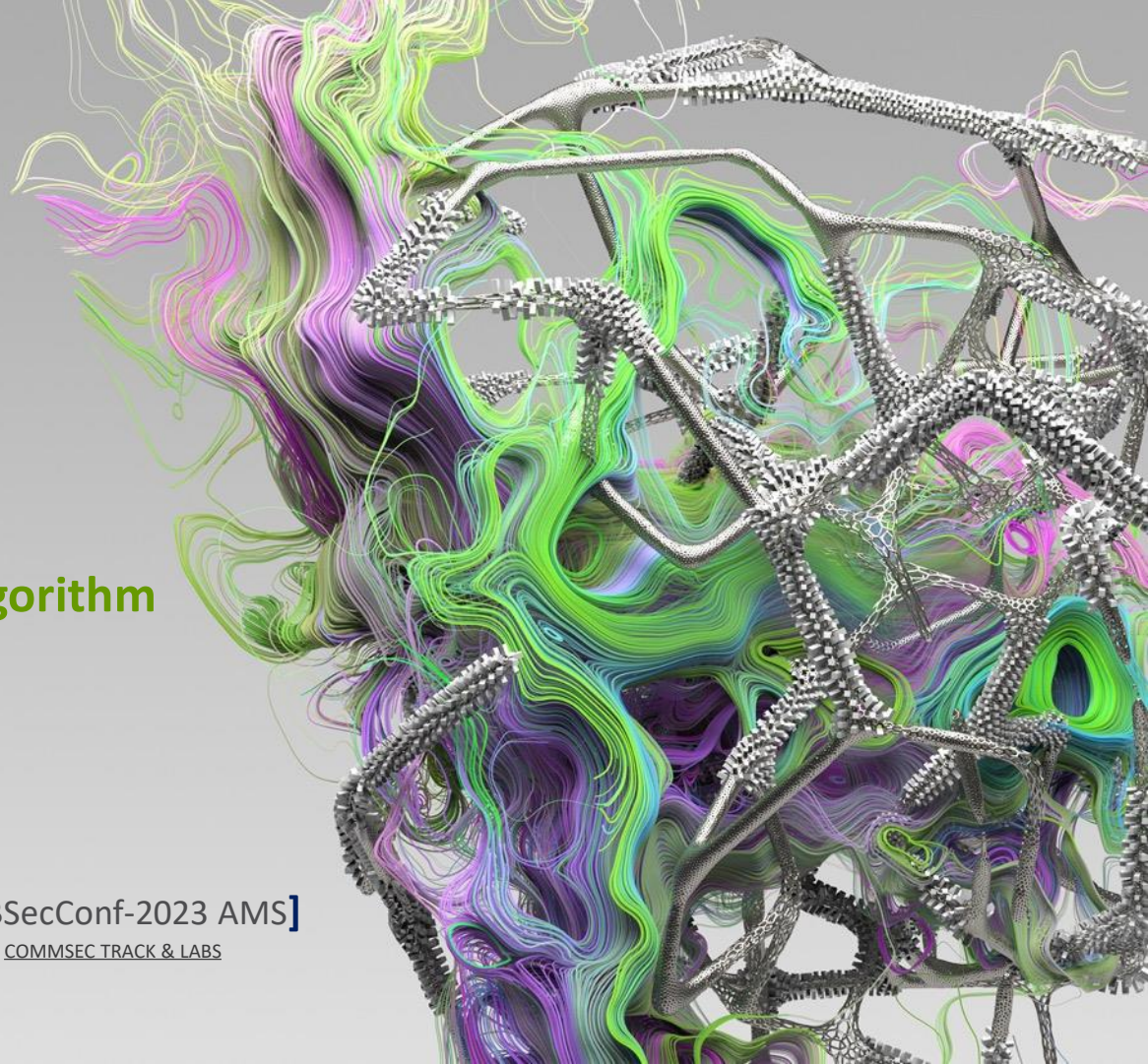


Exploring JARM

An Active TLS Fingerprinting Algorithm

[Mohamad Mokbel | @MFMokbel | HITBSecConf-2023 AMS]

COMMSEC TRACK & LABS

April 20, 2023 - Amsterdam

Biography

- Senior Security Researcher at Trend Micro
  - Member of the Digital Vaccine (DV) Lab
- Interests:
  - RE, Malware Research,
  - IDS/IPS,
  - C++, Compilers & Software Performance Analysis,
  - Exotic Communication Protocols

TLS/SSL Handshake

- Right after the TCP handshake, the client sends the TLS Client Hello packet, and the server Ack's the packet, followed by sending the TLS Server Hello packet.

- It is important to note that these packets are not encrypted for any version of the TLS/SSL protocol.
  - You can post-process those packets off of a packet capture or simply, a binary file.

Client Hello Packet/Message

```
Transport Layer Security
 TLSv1.2 Record Layer: Handshake Protocol: Client Hello
   Content Type: Handshake (22)
   Version: TLS 1.2 (0x0303)
   Length: 419
   Handshake Protocol: Client Hello
       Handshake Type: Client Hello (1)
       Length: 415
       Version: TLS 1.2 (0x0303)
       Random: 3dbc7646f6459f15b5e18567ac0e79dffc4107c2c6af478a4b1fcdc5482cc857
       Session ID Length: 32
       Session ID: 91cf45ab4f28a85565ef16ca92bcb0629b0be368dcd73867c40bc380ae81234e
       Cipher Suites Length: 138
       Cipher Suites (69 suites)
       Compression Methods Length: 1
       Compression Methods (1 method)
```

1

```
Extensions Length: 204
     Extension: server_name (len=15)
     Extension: extended_master_secret (len=0)
     Extension: max_fragment_length (len=1)
     Extension: renegotiation_info (len=1)
     Extension: supported_groups (len=10)
     Extension: ec_point_formats (len=2)
     Extension: session_ticket (len=0)
     Extension: application_layer_protocol_negotiation (len=60)
     Extension: signature_algorithms (len=20)
     Extension: key_share (len=38)
     Extension: psk_key_exchange_modes (len=2)
     Extension: supported_versions (len=7)
```

2

Server Hello Packet/Message

```
Transport Layer Security
 TLSv1.2 Record Layer: Handshake Protocol: Server Hello
   Content Type: Handshake (22)
   Version: TLS 1.2 (0x0303)
   Length: 72
   Handshake Protocol: Server Hello
       Handshake Type: Server Hello (2)
       Length: 68
       Version: TLS 1.2 (0x0303)
       Random: 626813ad56814273a072736b63f41e85b444fbc9faedc0c0444f574e47524401
       Session ID Length: 0
       Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
       Compression Method: null (0)
       Extensions Length: 28
       Extension: extended_master_secret (len=0)
       Extension: renegotiation_info (len=1)
       Extension: ec_point_formats (len=2)
       Extension: session_ticket (len=0)
       Extension: application_layer_protocol_negotiation (len=5)
```

In a Nutshell

- JARM is an active TLS fingerprinting algorithm vs the passive JA3 (TLSVersion,Ciphers,Extensions,EllipticCurves,EllipticCurvePointFormats) and JA3S (TLSVersion,Cipher,Extensions).
- It works by sending specially crafted 10 TLS Client Hello requests, with different options, probing the server for specific TLS Server Hello messages.
  - Cipher, TLS minor version and list of extensions.
    - All the requests' responses, including Cipher and TLS minor version, are fuzzy hashed, and the list of extensions is sha-256 hashed.
    - Finally, both hashes are concatenated to form the final 62-char JARM hash, against a given server.

Talk Layout

- Motivation
- Prior Work
- Types of Requests
- JARM RAW Fingerprint (intermediate representation)
- JARM Hybrid Fuzzy Hash (Cipher Suite and Version)
- Examples/Interesting Fingerprints
- Decode/Demangle a JARM hash
- Why and Oddities
- Demo

Prior Work

- Salesforce (Easily Identify Malicious Servers on the Internet with JARM)
  - https://engineering.salesforce.com/easily-identify-malicious-servers-on-the-internet-with-jarm-e095edac525a/
- Multiple vendors use it such as VT, Shodan, BinaryEdge, Censys, GitHub users,…
- JARM Randomizer: Evading JARM Fingerprinting
  - By Dagmawi Mulugeta
  - Presented at HITBAMS 2021
  - https://github.com/netskopeoss/jarm_randomizer
- PyJARM, a library for doing JARM fingerprinting using python (PaloAlto Networks)
  - https://github.com/PaloAltoNetworks/pyjarm
- JARM-Go by HDMoore
  - https://github.com/hdm/jarm-go
- Works like a JARM by Sketchymoose
  - https://github.com/sketchymoose/workslikeaJARM
  - https://sketchymoose.blogspot.com/2023/01/they-are-always-after-me-lucky-jarms.html

TLS Client Hello Packets – 10 Types

```
enum packet_type
 {
  /* 01 */ tls_1_2_forward,
  /* 02 */ tls_1_2_reverse,
  /* 03 */ tls_1_2_top_half,
  /* 04 */ tls_1_2_bottom_half,
  /* 05 */ tls_1_2_middle_out,
  /* 06 */ tls_1_1_middle_out,
  /* 07 */ tls_1_3_forward,
  /* 08 */ tls_1_3_reverse,
  /* 09 */ tls_1_3_invalid,
  /* 10 */ tls_1_3_middle_out
 };
```

- All share the same packet prologue, except for the different TLS version. Up to the Compression Methods().
- The differences are in the list of Ciphers chosen
- It is the Extensions list and specifics that changes per packet type.
  - 05/10: grease_extension()
  - 03/04/05/06: supported_versions_extension()

```
server_name_extension();
extended_master_key_extension();
max_frag_len_extension();
renegotiation_info_extension();
supported_groups_extension();
ec_point_formats_extension();
session_ticket_extension();
app_layer_proto_negotiation_extension(pktype);
    Order and nb of alpn protocols change
signature_algorithms_extension();
key_share_extension(pktype); // add grease() 05/10
psk_key_exchange_modes_extension();
```

JARM – Raw Fingerprint

`<cipher_suite/16-bit>|<tls_version/16-bit>|<alpn_ext/ascii>|<extension_type_x/16-bit>- ... -<extension_z>`

JARM (google.com [142.251.33.174]:443): 27d40d40d29d40d1dc42d43d00041d4689ee210389f4f6b4b5b1b93f92252d

```
 1 tls_1_2_forward     -> c02b|0303|h2|0017-ff01-000b-0023-0010
 2 tls_1_2_reverse     -> cca9|0303|h2|0017-ff01-000b-0023-0010
 3 tls_1_2_top_half    -> cca9|0303|h2|0017-ff01-000b-0023-0010
 4 tls_1_2_bottom_half -> c02f|0303||0017-ff01-000b-0023
 5 tls_1_2_middle_out  -> cca9|0303||0017-ff01-000b-0023
 6 tls_1_1_middle_out  -> c009|0302|h2|0017-ff01-000b-0023-0010
 7 tls_1_3_forward     -> 1302|0303||0033-002b
 8 tls_1_3_reverse     -> 1303|0303||0033-002b
 9 tls_1_3_invalid     -> |||
10 tls_1_3_middle_out  -> 1301|0303||0033-002b
```

```
c02b -> 27
0303 -> d
cca9 -> 40
0303 -> d
…
```

ext = sha256(h20017-ff01-000b-0023-0010h20017-ff01-000b-0023-0010h20017-ff01-000b-0023-00100017-ff01-000b-00230017-ff01-000b-0023h20017-ff01-000b-0023-00100033-002b0033-002b0033-002b)

TREND MICRO

JARM – Raw – Fuzzy Hash – Cipher Suite

```
std::string func::fuzzy_hash::get_cipher_bytes(const std::string& cb)
{
	if (cb.empty())
		return "00";
	else
	{
		std::vector<std::string> cipher_suite = { "0004",... ,"1305" }; // 69

		auto fcb = std::find(std::begin(cipher_suite), std::end(cipher_suite), cb);
		if (fcb != std::end(cipher_suite))
		{
			std::uint8_t cbidx = std::distance(std::begin(cipher_suite), fcb) + 1;

			std::stringstream idxb;
			idxb << std::hex << std::setfill('0') << std::setw(2) << +cbidx;

			return idxb.str();
		}
		else
			return "00";
	}
}
```

JARM – Fuzzy Hash - Version

```
std::string func::fuzzy_hash::get_version_bytes(const std::string& version)
{
     if (ver.empty())
     {
          return "0";
     }
     else
     {
          std::string options = "abcdef";
          int count = ver.back() - '0'; // tls minor version

          if (count > 6)
               return "0";
          else
               return std::string(1, options.at(count));
     }
}
```

```
a -> SSLv3   0x00
b -> TLSv1.0 0x01
c -> TLSv1.1 0x02
d -> TLSv1.2 0x03
e -> TLSv1.3 0x04
f -> TLSv1.4 0x05 (unassigned)
```

JARM – C&C Hashes

C2	JARM Fingerprint	Match (S/W/A)
Trickbot	22b22b09b22b22b22b22b22b22b22b352842cd5d6b0278445702035e06875c	0
AsyncRAT	1dd40d40d00040d1dc1dd40d1dd40d3df2d6a0c2caaa0dc59908f0d3602943	0
Metasploit	07d14d16d21d21d00042d43d000000aa99ce74e2c6d013c745aa52b5cc042d	0 (1.0K/VT-d) (1.2K/VT-ip)
Cobalt Strike	07d14d16d21d21d07c42d41d00041d24a458a375eef0c576d23a7bab9a9fb1	0 (277.0K/VT-d) (4.4K/VT-ip)
Merlin C2	29d21b20d29d29d21c41d21b21b41d494e0df9532e75299f15ba73156cee38	3_wp / 303_alexa (450k/VT) (10,760/Shodan)

JARM – Demangle/Decode Hash (Trickbot)

22b22b09b22b22b22b22b22b22b352842cd5d6b0278445702035e06875c

```
 1 tls_1_2_forward      -> C: 0xc014 (TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA) - V: 1 (TLSv1.0)
 2 tls_1_2_reverse      -> C: 0xc014 (TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA) - V: 1 (TLSv1.0)
 3 tls_1_2_top_half     -> C: 0x0039 (TLS_DHE_RSA_WITH_AES_256_CBC_SHA)   - V: 1 (TLSv1.0)
 4 tls_1_2_bottom_half  -> C: 0xc014 (TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA) - V: 1 (TLSv1.0)
 5 tls_1_2_middle_out   -> C: 0xc014 (TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA) - V: 1 (TLSv1.0)
 6 tls_1_1_middle_out   -> C: 0xc014 (TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA) - V: 1 (TLSv1.0)
 7 tls_1_3_forward      -> C: 0xc014 (TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA) - V: 1 (TLSv1.0)
 8 tls_1_3_reverse      -> C: 0xc014 (TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA) - V: 1 (TLSv1.0)
 9 tls_1_3_invalid      -> C: 0xc014 (TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA) - V: 1 (TLSv1.0)
10 tls_1_3_middle_out   -> C: 0xc014 (TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA) - V: 1 (TLSv1.0)
```

Extensions sha-256 hash: 352842cd5d6b0278445702035e06875c

JARM – Demangle/Decode Hash (WP)

27d3ed3ed0003ed1dc42d43d00041d6183ff1bfae51ebd88d70384363d525c

This hash is shared by 19292/9055 WP site. 0 on Shodan. FE server is Cloudflare.

```
 1 tls_1_2_forward      -> C: 0xc02b (TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256-R)                          - V: 3 (TLSv1.2)
 2 tls_1_2_reverse      -> C: 0xcc14 (IANA_Unassigned_TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256_OLD-R) - V: 3 (TLSv1.2)
 3 tls_1_2_top_half     -> C: 0xcc14 (IANA_Unassigned_TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256_OLD-R) - V: 3 (TLSv1.2)
 4 tls_1_2_bottom_half  -> C: no value                                                                    - V: no value
 5 tls_1_2_middle_out   -> C: 0xcc14 (IANA_Unassigned_TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256_OLD-R) - V: 3 (TLSv1.2)
 6 tls_1_1_middle_out   -> C: 0xc009 (TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA-W)                            - V: 2 (TLSv1.1)
 7 tls_1_3_forward      -> C: 0x1302 (TLS_AES_256_GCM_SHA384-R)                                           - V: 3 (TLSv1.2)
 8 tls_1_3_reverse      -> C: 0x1303 (TLS_CHACHA20_POLY1305_SHA256-R)                                     - V: 3 (TLSv1.2)
 9 tls_1_3_invalid      -> C: no value                                                                    - V: no value
10 tls_1_3_middle_out   -> C: 0x1301 (TLS_AES_128_GCM_SHA256-R)                                           - V: 3 (TLSv1.2)

Extensions sha-256 hash: 6183ff1bfae51ebd88d70384363d525c
```

JARM – Fingerprint (xda-developers.com [104.18.19.88]:443)

27d3ed3ed0003ed1dc42d43d00041d6183ff1bfae51ebd88d70384363d525c

```
 1 tls_1_2_forward     -> c02b|0303|h2|0000-0017-ff01-000b-0023-0010
 2 tls_1_2_reverse     -> cc14|0303|h2|0000-0017-ff01-000b-0023-0010
 3 tls_1_2_top_half    -> cc14|0303|h2|0000-0017-ff01-000b-0023-0010
 4 tls_1_2_bottom_half -> |||
 5 tls_1_2_middle_out  -> cc14|0303||0000-0017-ff01-000b-0023
 6 tls_1_1_middle_out  -> c009|0302|h2|0000-0017-ff01-000b-0023-0010
 7 tls_1_3_forward     -> 1302|0303||0033-002b
 8 tls_1_3_reverse     -> 1303|0303||0033-002b
 9 tls_1_3_invalid     -> |||
10 tls_1_3_middle_out  -> 1301|0303||0033-002b
```

JARM – Fingerprint (99.86.237.136 - Shodan)

29d29d00029d29d21c41d41d00041dba71dd2df645850cf5f0b5af18a5fdcf

```
 1 tls_1_2_forward      -> c02f|0303||000b-ff01-0023
 2 tls_1_2_reverse      -> c02f|0303||000b-ff01-0023
 3 tls_1_2_top_half     -> |||
 4 tls_1_2_bottom_half  -> c02f|0303||000b-ff01-0023
 5 tls_1_2_middle_out   -> c02f|0303||000b-ff01-0023
 6 tls_1_1_middle_out   -> c013|0302||000b-ff01-0023
 7 tls_1_3_forward      -> 1301|0303||002b-0033
 8 tls_1_3_reverse      -> 1301|0303||002b-0033
 9 tls_1_3_invalid      -> |||
10 tls_1_3_middle_out   -> 1301|0303||002b-0033
```

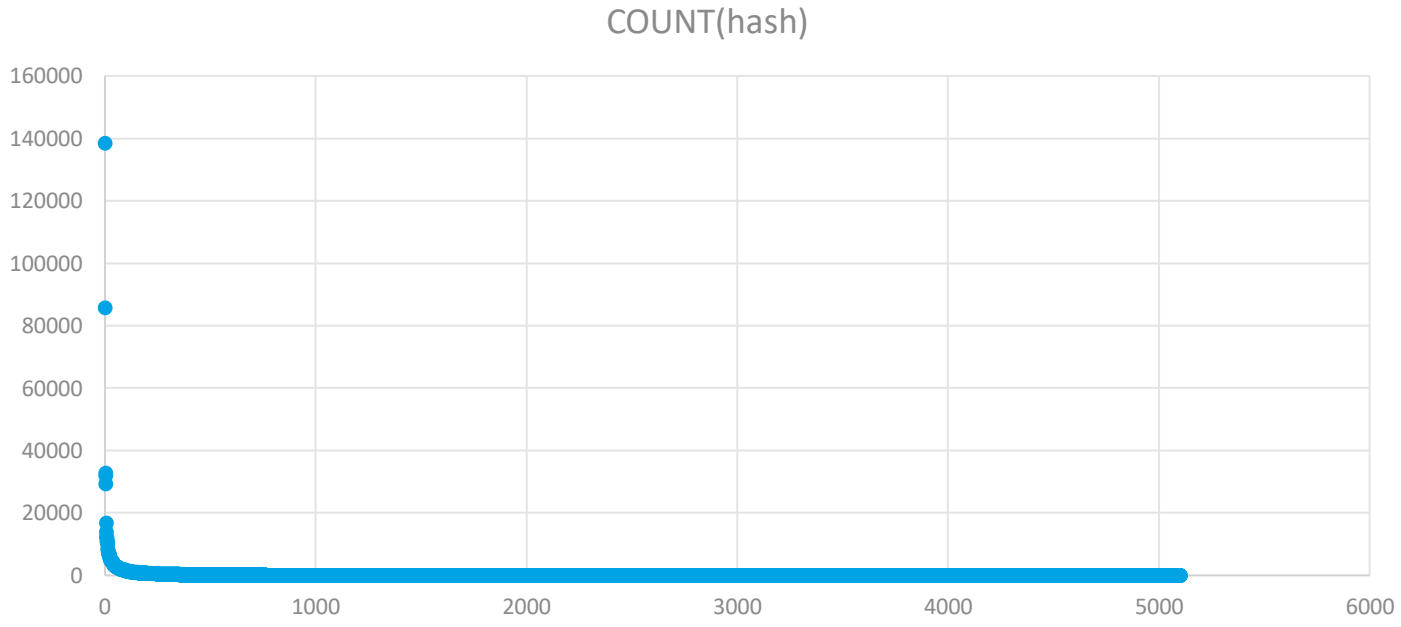
As of May 31, 2022, this hash is shared by 11,980,624 servers. FE server is CloudFront.

JARM – Alex Top 1-Million Site

- Total of 5001 unique hashes (957550/1m resolved successfully at the time of scanning)
  - 27d3ed3ed0003ed1dc42d43d00041d6183ff1bfae51ebd88d70384363d525c (total: 138475)
  - 21d10d00021d21d21c21d10d21d21dfa4f2d467cfc282d8a9029b2af1af43b (total: 997)
  - 2ad2ad16d2ad2ad22c42d42d000000d342d5966a57139eeaff9f8bc4841b25 (total: 2)
  - 40d40d40d3fd40d00042d42d00000045613aa8a1719a3ecc168186fa9ed346 (total: 2)
  - 02d02d20d02d02d02c02d02d02d02d4907d96b59558a84506cb23b33dad7ae (total: 2)
    - (btsnetops.com, flymna.com)

```
 1 tls_1_2_forward      -> C: 0xc030 (TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384)  - V: 3 (TLSv1.2)
 2 tls_1_2_reverse      -> C: 0xc030 (TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384)  - V: 3 (TLSv1.2)
 3 tls_1_2_top_half     -> C: 0x009f (TLS_DHE_RSA_WITH_AES_256_GCM_SHA384)    - V: 3 (TLSv1.2)
 4 tls_1_2_bottom_half  -> C: 0xc030 (TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384)  - V: 3 (TLSv1.2)
 5 tls_1_2_middle_out   -> C: 0xc030 (TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384)  - V: 3 (TLSv1.2)
 6 tls_1_1_middle_out   -> C: 0xc014 (TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA)     - V: 2 (TLSv1.1)
 7 tls_1_3_forward      -> C: 0x1302 (TLS_AES_256_GCM_SHA384)                 - V: 3 (TLSv1.2)
 8 tls_1_3_reverse      -> C: 0x1302 (TLS_AES_256_GCM_SHA384)                 - V: 3 (TLSv1.2)
 9 tls_1_3_invalid      -> C: no value                                        - V: no value
10 tls_1_3_middle_out   -> C: no value                                        - V: no value
```

JARM – Alex Top 1-Million Site



JARM – Interesting Fingerprints (Insecure Ciphers)

- semenindonesia.com (WP 100k)
  - 01d01d20d01d01d01c01d01d01d01dbcb145a8c2a715eeaf337d00ba95d886
  - 01 -> TLS_RSA_WITH_RC4_128_MD5 (insecure cipher suite)
  - d/3 -> TLSv1.2
  - Shared with 12 domains in the Alexa top-1m
- europlayers.com (WP 100k)
  - 02d02d00002d02d02c02d02d02d02dd3b67dd3674d9af9dd91c1955a35d0e9
  - 02 -> TLS_RSA_WITH_RC4_128_SHA (insecure cipher suite)
  - d/3 -> TLSv1.2
  - Shared with 19 domains in the Alexa top-1m
- 55 domains from Alexa top-1m domains use the cipher TLS_RSA_WITH_RC4_128_MD5
- 149 domains from Alexa top-1m domains use the cipher TLS_RSA_WITH_RC4_128_SHA

JARM – Interesting Fingerprints (TLS v1.0)

- 842 domains from Alexa top-1m domains use TLS version 1.0, coupled with weak ciphers
- 47 domains from WP's top 100k domains use TLS version 1.0, coupled with weak and insecure ciphers
- 1 domain from WP's top 100k domains uses SSL version 3.0, coupled with a weak cipher
  - barbacena.com.br (Business and Economy)
- 2 domains from Alexa top-1m domains use SSL version 3.0, coupled with weak and insecure ciphers
  - catastroweb.com.ar
  - cac.lv

JARM – What Does it Mean

- Many factors are at play that determine the final hash of a server, including but not limited to:
  - Specific library/ies used and their versions, OS version and platform, order, default settings, and configuration.
  - What TLS version does the server support, 1.1, 1.2, 1.3? (*supported_versions* extension)
  - Will the server accepts a TLS 1.2 cipher in a TLS 1.3 request?
  - Order of ciphers (ex., weakest to strongest); which one will the server choose?
  - ALPN extension (presence and order), GREASE extension,…
- If multiple servers share the same first 30 chars, it means they have similar configurations(!), but not exactly the same given that the extensions support is different.

JARM – Oddities

- Some servers return a different JARM hash every time you query it.
  - For example, the "in.gr" server, sometimes returns the alpn extension, and sometimes doesn’t!
  - This changes the fidelity rate in the hash.
- For the same server, you could have different hashes at different times.
  - For some servers, not all Client Hello requests are accepted consistently, and this changes the hash from time to time.

JARM – Demo

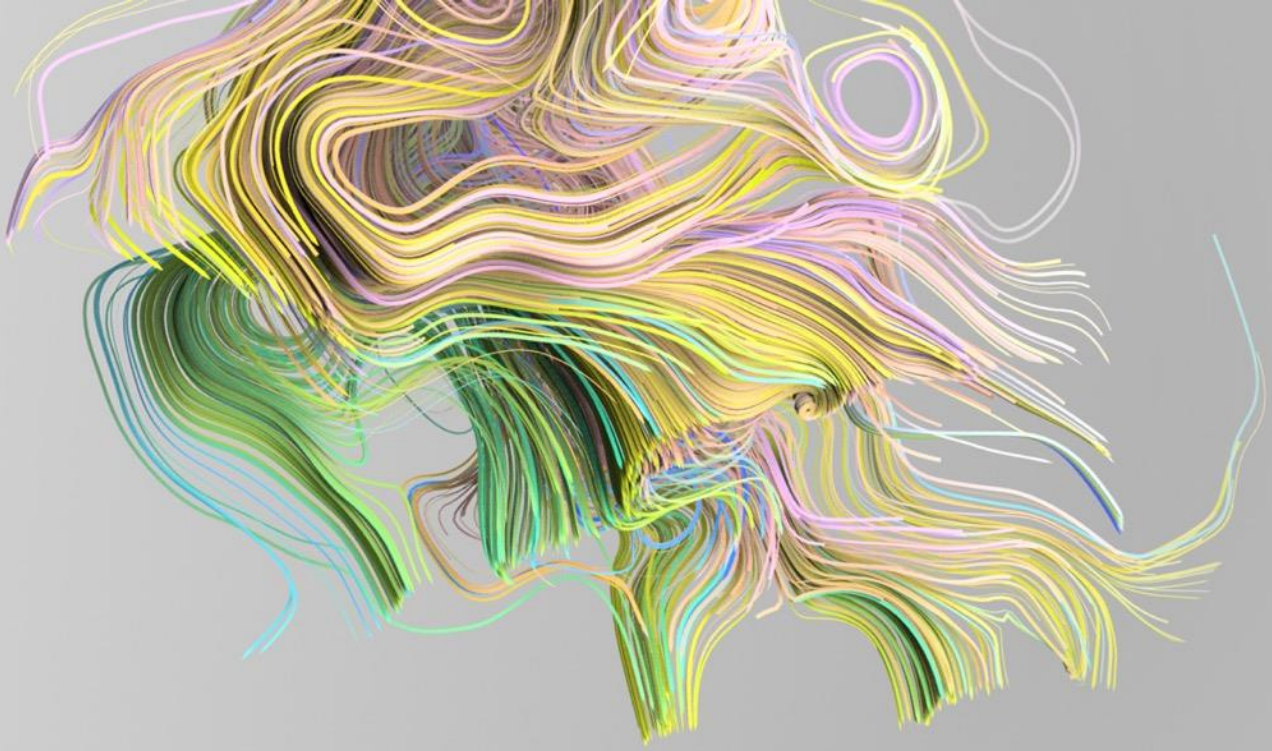
JARM-CPP

Third-Party Libraries

- Hash-Library: for SHA-256
- SFML (network module): for socket communications
  - Issues!
- Color Console: for console colouring
- cxxopts: for parsing command-line arguments

Statistics

JARM Hash	Count (WP/Shodan(May 31, 2022))
27d3ed3ed0003ed1dc42d43d00041d6183ff1bfae51ebd88d70384363d525c	19292/236,709
27d3ed3ed0003ed00042d43d00041df04c41293ba84f6efe3a613b22f983e6	3396/52,929
29d29d00029d29d00042d43d00041dd469afa8cfbe5e42c631eb3fc55d6787	2896/92,355
29d29d15d29d29d00042d42d000000038eaaf490bec8dc33757f165ce01762	2840/154,644
2ad2ad0002ad2ad00042d42d0000005d86ccb1a0567e012264097a0315d7a7	2787/126,536
3fd3fd07d3fd3fd00042d42d0000005fd00fabd213a5ac89229012f70afd5c	1705/57,624
29d3fd00029d29d00042d43d27d0003d5888b882bff12119feb529a94aa241	1507/1,829
07d14d16d21d21d07c07d14d07d21df81841108a56803289beb36a0dd595dc (andaluciaesdigital.es, andaluciaemprende.es) Server: cisco-IOS, micro_httpd, PHP/7.4.6	2/2,689



Conclusion

Thank You

Q & A

So, what’s JARM?