TLS Fingerprinting

JA3 and JA3S

سارة محمد آل جابر

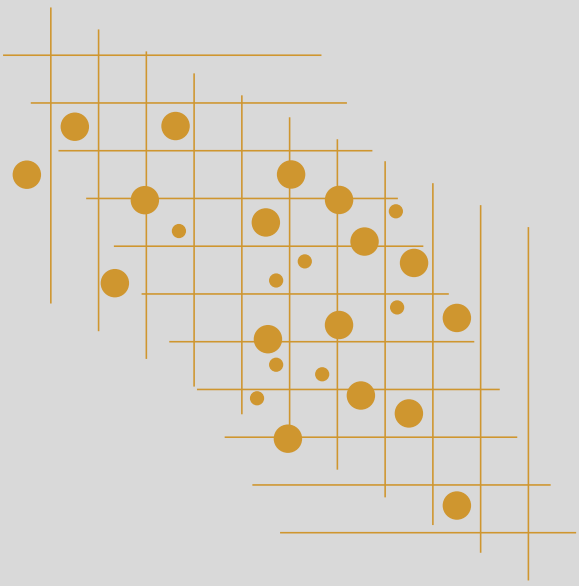
2021/02/20

Twitter: s4o_o

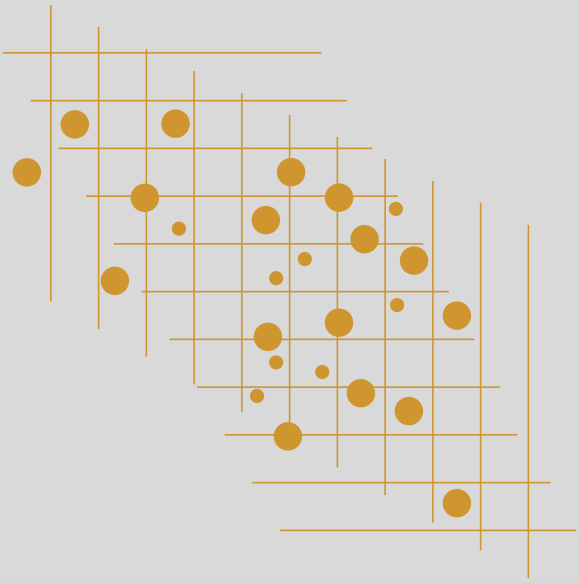
شكر وتقدير

كل الشكر موصول للأستاذ علي الوشلي على دعمة المستمر وتدقيق ومراجعة هذا العمل لإخراجه بهذا الشكل.

Twitter Account: @ali_alwashali



الفهرس



فهرس الصور

المقدمة

التحليل الجنائي للشبكات

تحليل الشبكات هو أحد فروع التحليل الجنائي الرقمي التي تساعد على استخراج معلومات مهمة تساعد المحلل على معرفة فيما إذا كان هناك أي مشاكل في الشبكة أو إذا كان هناك اختراق للشبكة ويمكن تقسيم عملية تحليل الشبكات على مرحلتين، المرحلة الأولى مرحلة التقاط الحزم والبيانات والمرحلة الثانية مرحلة التحليل ويمكن تعريف المرحلة الثانية بأنها التمكن من قراءة Packet Traffic (حزم البيانات المارة بالشبكات)، ومعرفة تفاصيلها لأسباب أمنية بالنسبة لمختصين أمن المعلومات أو اختبار وضع الشبكة وأدائها ب النسبة لمختصين الشبكات.

من خلال قراءة وتحليل حزم البيانات من الممكن التعرف على الآتي:

- أداء الشبكة.
- اكتشاف الهجمات التي قد تحدث ب الشبكة.
- اكتشاف البرمجيات الضارة.
- معرفة ما الذي يحدث ب التحديد.
- حل المشاكل.

ما هو Transport Layer Security؟

كما هو معروف تعتمد الشبكات بنقل البينات على ما يسمى ب البروتكولات و TLS هو أحد بروتكولات التشفير الذي يمكننا من توفير اتصال أمن بين الطرفين حيث يقوم بتشفير البينات المرسلة لمنع أي طرف أخر غير معني ب قراءة هذه البيانات أو تعديلها.

يتم استخدام TLS في تطبيقات الخادم والعميل و Client/Server Application سواء كانت متصفحات الانترنت أو برامج المراسلة أو أي تطبيق أو برنامج يعتمد على التواصل بين .Client/Server communication

TLS Handshake

آلية عمل TLS تبدأ باتصال العميل ب الوكيل من خلال ارسال ما يسمى ب المصافحة (Three Way Handshake) التي من خلالها يقوم العميل بإرسال مجموعة قائمة التشفير التي يمكنه استخدامها وتسمى (Cipher suite) عندما تصل القائمة الى الخادم يقوم هو بدورة بالبحث عن طريقة للتشفير تتوافق مع العميل وارسالها للعميل Server Hello بعد ذلك يقوم بإرسال Public key certificate والتي تحتوي على بعض معلومات الخادم مثل الاسم وCertificate Authorities والتي تعبر على عن الجهة التي قامت بإصدار الشهادة وتوثيقها.

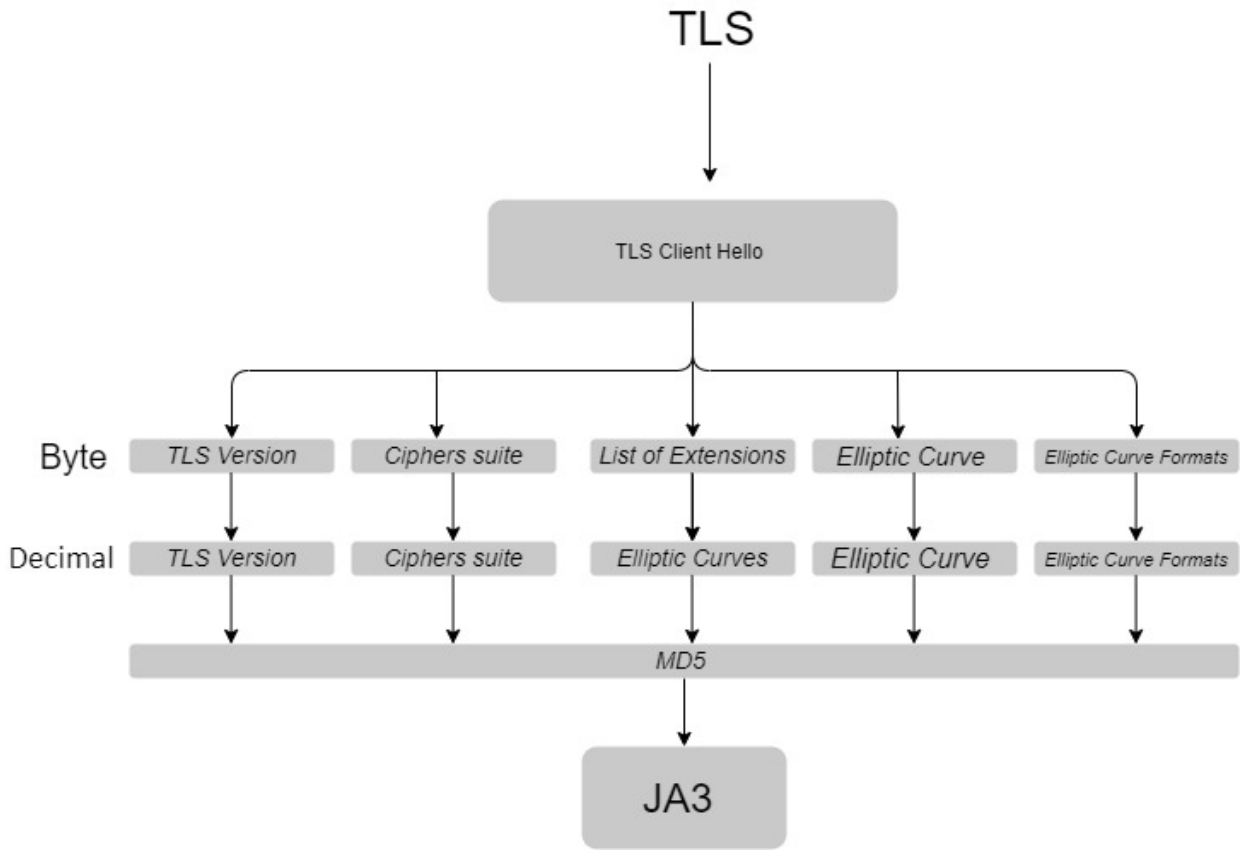
JA3S وJA3

عملية حسابية يتم الاستعانة بها ببعض القيم المتواجدة ب TLS client hello او TLS Server hello لتحديد قيمة عالية الدقة ثابتة بجهاز الوكيل وبصمة خاصة بجهاز الخادم.

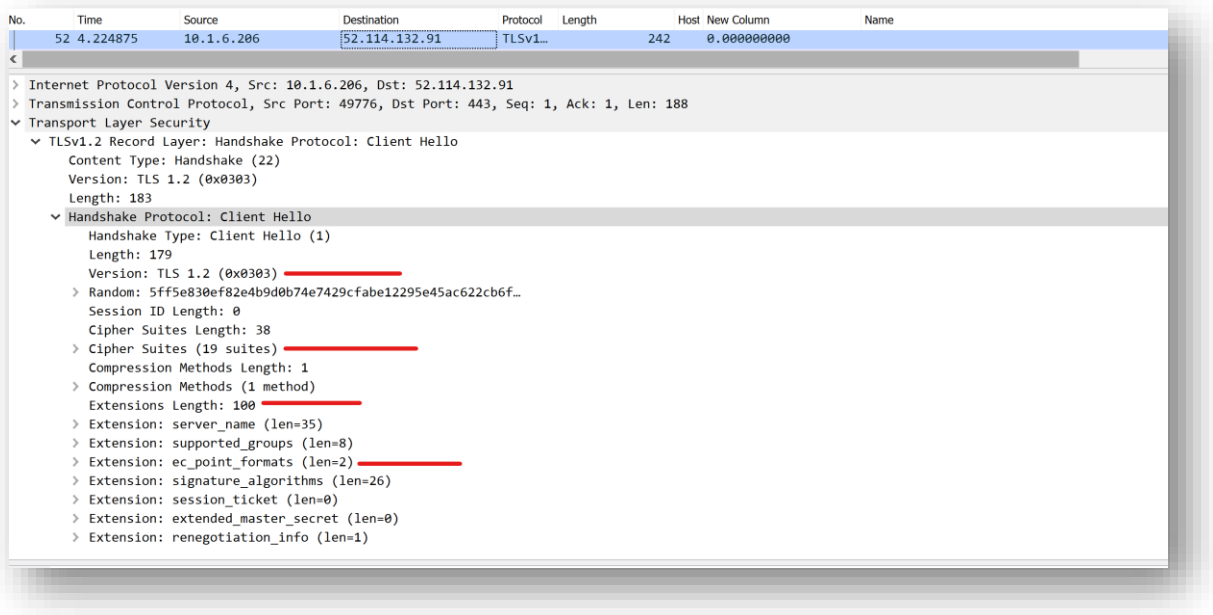
ظهر مفهوم TLS fingerprint كأول مره في عام 2015 من الباحث leE Brotherston [1]

JA3

عملية حسابية يتم فيها تحويل بعض من محتويات القيمة الجزئية (Byte) ل Client hello لقيمة عشرية (Decimal) من ثم حساب MD5 وذلك حتى نحصل على قيمة ثابته دائما ب 32 بت.



في المثال التالي نقوم بحساب قيمة JA3 ل Packet Traffic البرمجية الخبيثة Emotet



صورة 1: مثال ل Client Hello من Wireshark

1- بداية نقوم ب تحديد المحتويات المراد حسابها وهي كالتالي وب الترتيب

Version, Ciphers suites, List of Extensions, Elliptic Curves, and Elliptic Curve Formats

2- نقوم بحساب القيمة العشرية لها لتصبح ك التالي:

771,49196-49195-49200-49199-49188-49187-49192-49191-49162-49161-49172-49171-157-156-61-60-53-47-10,0-10-11-13-35-23-65281,29-23-24,0

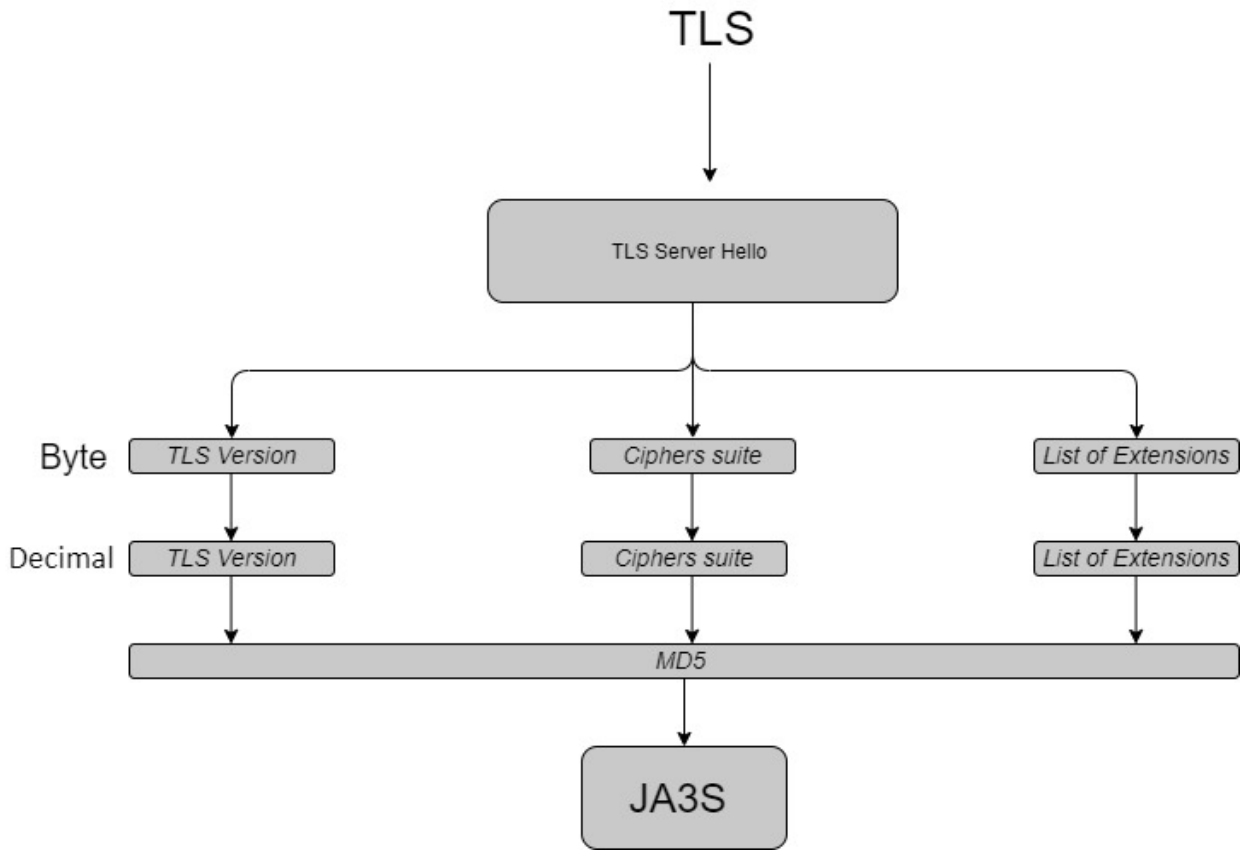
3- الان نقوم بحساب MD5 لهذه القيمة ليصبح الناتج النهائي:

37f463bf4616ecd445d4a1937da06e19

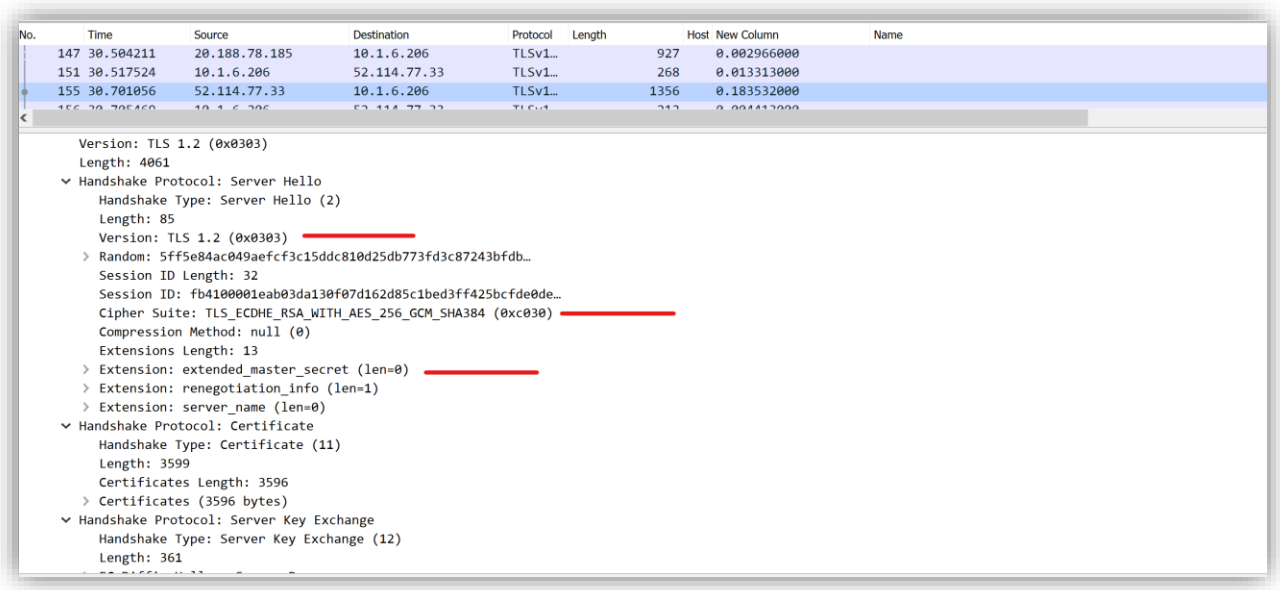
تأتي أهمية حساب MD5 في حال احتوت TLS client Hello على عدد كبير من Extensions يمكن دائما وبالنهاية الحصول على قائمة ثابتة تحتوي على 32 بت.

JA3S

عملية حسابية يتم فيها تحويل بعض من محتويات القيمة الجزئية Server hello (Byte) لقيمة عشرية (Decimal) ومن ثم حساب MD5



بالمثال التالي نستعرض طريقة حساب JA3S



صورة 2 : مثال ل Server Hello من Wireshark

1- بداية نقوم ب تحديد المحتويات المراد حسابها وهي كالتالي وب الترتيب

Version, Ciphers suites, List of Extensions

2- نقوم بحساب القيمة العشرية لها لتصبح ك التالي:

771,CipherSuite(TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384),16-23-65281-0

3- الان نقوم بحساب MD5 لهذه القيمة ليصبح الناتج النهائي:

6f333ffd93108d65b2b61ff0a061a0d1

في المثالين السباقين قمت ب الاستعانة ب أداة pyja3 و أداة pyja3s تم تطويرها من قبل John Althouse [2]

كيفية عمل الأداة

في البداية نستعرض الخيارات المتاحة في الأداة:

```
$ python ja3.py -h
usage: ja3.py [-h] [-a] [-j] [-r] pcap

A python script for extracting JA3 fingerprints from PCAP files

positional arguments:
  pcap            The pcap file to process

optional arguments:
  -h, --help      show this help message and exit
  -a, --any_port  Look for client hellos on any port instead of just 443
  -j, --json      Print out as JSON records for downstream parsing
  -r, --research  Print packet related data for research (json only)
```

صورة 3 : أداة ja3.py

1. JA3

لاستخراج قيمة JA3 باستخدام أداة ja3.py نقوم بكتابة الأمر التالي:

Ja3.py –json pcapfile.pcp

```
$ python ja3.py --json Example-1-2021-01-06-Emotet-infection.pcap
[
    {
        "destination_ip": "52.114.132.91",
        "destination_port": 443,
        "ja3": "771,49196-49195-49200-49199-49188-49187-49192-49191-49162-49161-49172-49171-157-1
56-61-60-53-47-10,0-10-11-13-35-23-65281,29-23-24,0",
        "ja3_digest": "37f463bf4616ecd445d4a1937da06e19",
        "source_ip": "10.1.6.206",
        "source_port": 49776,
        "timestamp": 1609951279.80798
    },
```

صورة 4 : أداة ja3.py

JA3S .2

لاستخراج قيمة JA3S باستخدام أداة ja3s.py نقوم بكتابة الأمر التالي:
Ja3s.py –json pcapfile.pcp

```
$ python ja3s.py --json Example-1-2021-01-06-Emotet-infection.pcap
[
    {
        "destination_ip": "10.1.6.206",
        "destination_port": 49780,
        "ja3": "771,CipherSuite(TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384),16-23-65281-0",
        "ja3_digest": "6f333ffd93108d65b2b61ff0a061a0d1",
        "source_ip": "111.221.29.40",
        "source_port": 443,
        "timestamp": 1609951307.991171
    },
```

صورة 5: أداة ja3s.py

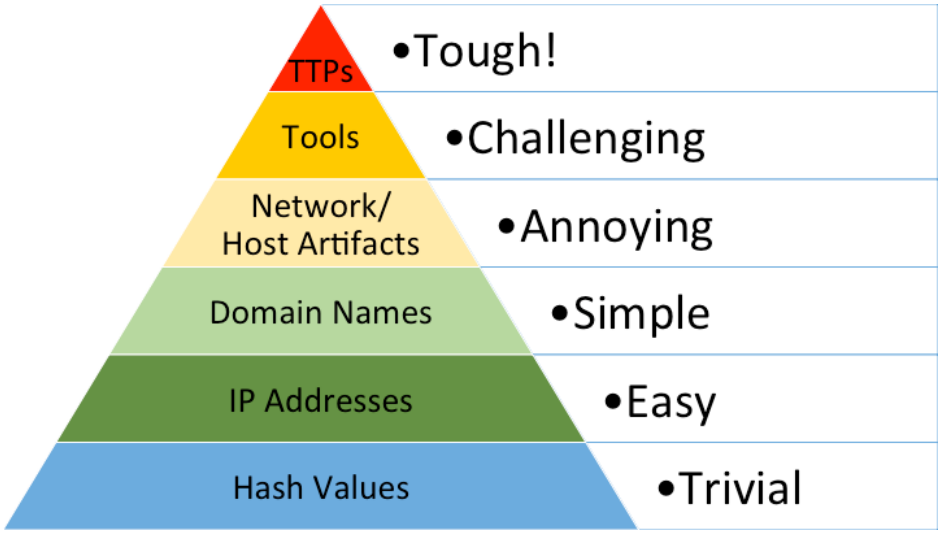
Threat Intelligence الطرق الاستخبارية لجمع المعلومات

نظرا للتحول الرقمي المستمر ولتنوع أشكال التكنولوجيا وتزايد عدد البرامج التطبيقات المستمرة فأن الهجمات بدورها أيضا بتزايد مستمر ودائم فتأتي هنا أهمية Threat Intelligence حيث تمكن الباحث من جمع وتحليل أهم أشكال البرمجيات الخبيثة حتى يتم التصدي لها بالشكل المطلوب.

يعتمد Threat Intelligence على التحليل والتنبه الدائم لأي جديد بعالم الهجمات لا استخراج اهم المعلومات التي تمكن المحلل او فريق أمن المعلومات من تصيد والتصدي لهذه الهجمات والتي تسمى ب مؤشرات الاختراق أو IOCs وتشكل خطوات Threat Intelligence شكل دائري ومستمر ابتدأ بجمع المعلومات معالجتها من ثم تحليلها.

هذه البيانات تتشكل في هرم يوضح أهمية وتواجد هذه المعلومات والذي يسمى ب The Pyramid of Pain أو هرم الألم.

مرتبة حسب مستوى سهولة تغير هذه المعلومة من قبل المخترق ابتدأ بالأسهل صعودًا حتى الأصعب.



صورة 6:The Pyramid of Pain

1. يحتوي المستوى الأول على قيم Hash والتي من السهل جدا تغيرها حيث ان تغير قيمة واحدة في البرمجية الخبيثة يقوم بتغير قميه Hash تمامًا
2. يأتي بالمستوى الثاني عنوان بروتكول الانترنت IP والذي من السهل تغيره أيضا من قبل المخترق حيث يمكنه بكل سهوله تغير واختيار عنوان جديد
3. أسماء النطاقات توازي عنوان الانترنت من ناحية السهولة حيث يسهل على المخترق أن يقوم بتغيرها بشكل مستمر
4. بالمستوى الرابع تأتي معلومات ال Network/Host Artifacts والتي تعد تحدي بالنسبة للمخترق لتغيرها وتعد من المعلومات المهم جمعها للتصيد لهذه الهجمات
5. يأتي بالمستوى الخامس الأدوات المستخدمة بالاختراق والتي تشكل أهمية كبرى لدى فريق أمن المعلومات حيث يصعب على المخترق تغيرها بشكل مستمر
6. أخيرا بالمستوى السادس TTPs Tactics, Techniques and Procedures والتي تعني تكتيكات واستراتيجيات الهجمة

والتي لا يمكن بسهولة تغيرها من قبل المخترق ويشكل هذه المستوى أهم وأقوى المعلومات في Threat Intelligence

JA3 و JA3S و Threat Intelligence

حيث أن هناك مجموعة من البرمجيات الخبيثة التي من الممكن أن تتطلب تواصل بين جهاز الضحية وجهاز المخترق C2 والذي بدورها تستخدم TLS تأتي أهمية JA3 و JA3S للتعرف على هذه الخوادم حيث أنه حتى في حالة تغير IPاو Domain سيبقى JA3 و JA3S ثابتان لا يتغيران.

ومن هنا تأتي أهميتها في Threat Intelligence وكما ذكرنا بالقسم السابق تعد معلومات Network/Host Artifacts من المعلومات القيم الحصول عليها للتصدي لهذه الهجمات حيث أن JA3 و JA3S من القيم التي من الصعب تغيرها.

وحيث أن الكثير من البرامج الان تدعم هذي الخاصة مثل (Bro,Darktrace,MISP,Moloch,NGiNXTrisul NSM) فمن السهل استخدامها لتحليل حزم الشبكات التي تحتوي على برمجية خبيثة واستخراج JA3 و JA3S وتضاف الى مؤشرات الاختراق IOCs

كما يوجد العديد من المصادر المفتوحة التي تحتوي على قائمة كبيرة ل JA3 و JA3S للاستعانة بها للتحقق من القيم التي قمنا باستخراجها ما اذا كانت لبرمجية خبيثة أو لا أشهرها ja3er.com

كما يمكن الاستفادة منها برسم Baseline خاص ببيئة معينة وتحديد قيم JA3 و JA3S وفي حال ظهور أي قيم جديدة من الممكن تحليلها والنظر فيها.

المراجع

1- https://blog.squarelemon.com/tls-fingerprinting/
2- https://engineering.salesforce.com/tls-fingerprinting-with-ja3-and-ja3s-247362855967
3- https://github.com/salesforce/ja3