Communication Security Enhancement in Application Layer Multicast Networks Group

Dr. Mothanna Alkubeily[•]
Rawan Khalifeh[••]

(Received 24 / 12 / 2024. Accepted 30 / 1 / 2025)

□ ABSTRACT □

Application-Layer Multicast (ALM) networks has designed to overcome the propagation issue of IP multicast, as they build a overlay tree composed of end-to-end unicast connections based on the cooperation of group members with each other. With the rise of cyber-attacks, group communication security and group key confidentiality in these networks has become critical to prevent unauthorized access to sensitive data, especially in military applications that rely on Drone-assisted VANET. This poses challenges in terms of building lightweight key management and distribution protocols, and achieving security when devices join or leave the group in dynamic environments.

This research aims to develop Arrows in a Quiver (AinQ), the key management and distribution protocol, to improve the security of group communication in different networks by minimizing key generation and rekeying time, especially in highly dynamic groups. Also, examining group key randomization as a mechanism that makes key prediction difficult for unauthorized parties, adding an additional layer of protection to group communications and reducing the chances of session compromise or message decryption.

Keywords: Application Layer Multicast, Group Communication Security, Drone-assisted VANET, Arrows in a Quiver protocol, Group Key Randomness.

[•] **Professor, Department of Communication and Electronics, Faculty of mechanical and electrical engineering, Latakia University, Lattakia, Syria. mothanna.alkubeily@gmail.com**

[••] **Master student, Department of Communication and Electronics, Faculty of mechanical and electrical engineering, Latakia University, Lattakia, Syria. rawan.khalifeh.1995@gmail.com**

تحسين أمن اتصال المجموعات في الشبكات التطبيقية متعددة البث

د. مثنى القبيلي •

روان خليفة••

(تاريخ الإيداع 24 / 12 / 2024. قُبِل للنشر في 30/ 1 / 2025)

□ ملخّص □

جاءت الشبكات التطبيقية متعددة البث (ALM) للتغلب على مشكلة النشر في البث المجموعاتي IP، حيث أنها تبني شجرة تغطية مؤلفة من اتصالات نهاية إلى نهاية أحادية البث اعتماداً على تعاون أعضاء المجموعة مع بعضهم البعض. ومع تزايد الهجمات السيبرانية، أصبح أمن اتصال المجموعات وضمان سرية مفتاح المجموعة في هذه الشبكات أمراً بالغ الأهمية لمنع الوصول غير المصرح به إلى البيانات الحساسة، وخاصة في التطبيقات العسكرية التي تعتمد على شبكات العربات بمساعدة الطائرات دون طيار. وهو ما يطرح تحديات فيما يتعلق ببناء بروتوكولات إدارة وتوزيع مفاتيح خفيفة الوزن تتناسب مع طبيعة الأجهزة، وتحافظ على الأمن عندما تغادر الأجهزة مجموعة أو تنضم إليها ضمن البيئات الديناميكية.

يهدف هذا البحث إلى تطوير بروتوكول Arrows in a Quiver (AinQ) لإدارة وتوزيع المفاتيح وقابليته للتوسع، بهدف تحسين أمن اتصال المجموعات في بيئات الشبكات المختلفة من خلال تخفيض زمن توليد وتحديث المفاتيح، خصوصاً في المجموعات ذات الديناميكية العالية. كما يتناول البحث دراسة عشوائية مفتاح المجموعة كآلية تجعل التنبؤ بالمفتاح صعباً على الأطراف غير المصرح لها، مما يضيف طبقة إضافية من الحماية للاتصالات الجماعية ويقلل فرص اختراق الجلسة أو فك تشفير الرسائل المتبادلة.

الكلمات المفتاحية: الشبكات التطبيقية متعددة البث، أمن اتصال المجموعات، شبكات العربات بمساعدة الطائرات بدون طيار، بروتوكول Arrows in a Quiver، عشوائية مفتاح المجموعة.

• **أستاذ، قسم هندسة الاتصالات والالكترونيات ، كلية الهندسة الميكانيكية والكهربائية، جامعة تشرين، اللاذقية سورية. mothanna.alkubeily@gmail.com**

•• **طالبة دراسات عليا (ماجستير) ، قسم هندسة الاتصالات والالكترونيات ، كلية الهندسة الميكانيكية والكهربائية، جامعة تشرين، اللاذقية سورية. rawan.khalifeh.1995@gmail.com**

مقدمة:

جاءت الشبكات التطبيقية متعددة البث (ALM) كحل للتحديات التي تواجه تقنية الإرسال المجموعاتي، حيث تقوم بإنشاء شجرة تغطية من المضيفين المشاركين في جلسة البث [1,2]. تعتمد هذه التقنية على اتصال المجموعة لتوزيع العقد على مجموعات ديناميكية يمكن أن تتغير بانضمام عقد جديدة أو مغادرة عقد أخرى أو حدوث فشل في بعضها. تسبب هذه الديناميكية تحديات أمنية، خاصة فيما يتعلق بسرية المعلومات المعتمدة على مفاتيح التشفير لتأمين البيانات المرسلة.

يعد أمن اتصال المجموعات في شبكات البث المتعدد في طبقة التطبيقات أمراً أساسياً نظراً للاعتماد المتزايد عليها، خاصة في تطبيقات إنترنت الأشياء (IoT: Internet of Things) وإنترنت المركبات (IoV: Internet of Vehicles)، حيث يتيح الاتصال المجموعاتي إرسال الرسائل بكفاءة لعدة مستخدمين [3,4]. مع انتشار الطائرات دون طيار في مجالات مثل النقل اللوجستي، والمراقبة، والاتصالات الطارئة، والعمليات العسكرية، ظهرت تحديات جديدة لأمن اتصالات المجموعة، خصوصاً في المهام التعاونية بين المركبات والطائرات دون طيار. في هذا السياق، يُعد تطوير بروتوكولات إدارة مفاتيح آمنة وموثوقة من الأولويات لضمان سرية وسلامة البيانات في الشبكات التطبيقية متعددة البث وحمايتها من التهديدات.

أهمية البحث وأهدافه:

تأتي أهمية هذا البحث من دور الشبكات التطبيقية متعددة البث في تحسين أداء التطبيقات التي تتطلب عرض حزمة واسع، مثل التعليم عن بعد والمؤتمرات الفيديوية، بفضل قدرتها على تخفيض استهلاك عرض الحزمة، مما يخفض من تكاليف البنية التحتية ويحسن أداء الشبكة. من ناحية أخرى، ومع تزايد الهجمات السيبرانية، برزت حماية اتصال المجموعات من الوصول غير المصرح به، والتي تستلزم توفير السرية والمصداقية وسلامة الرسائل ضمن الجلسة كتحدٍ رئيسي، حيث أن العضوية المفتوحة والمجهولة (إمكانية الانضمام والمغادرة لأي عقدة) تشكل خطراً على أمن اتصال المجموعة. تعد إدارة المفاتيح واحدة من أكثر القضايا صعوبة، ونتيجة لذلك، تم التركيز على موضوع إدارة وتوزيع المفاتيح وأهميته في الحفاظ على سرية الجلسة.

نهدف في هذا البحث إلى تطوير بروتوكول توليد وتوزيع المفاتيح Arrows in a Quiver (AinQ) وقابليته للتوسع لتعزيز أمن اتصال المجموعات في بيئات شبكات مختلفة، من خلال تخفيض تكاليف الوقت المرتبطة بتوليد وتحديث المفاتيح عند انضمام الأعضاء أو مغادرتهم خاصة في المجموعات ذات الديناميكية العالية. إضافة إلى دراسة عشوائية مفتاح المجموعة التي تجعل من الصعب على الأطراف غير المصرح لها التنبؤ أو استنتاج المفتاح، مما يوفر طبقة إضافية من الحماية لاتصالات المجموعات، ويقلل فرص اختراق الجلسة أو فك تشفير الرسائل المتبادلة.

طرائق البحث ومواده:

يستخدم المطورون لغة Python لسهولة تعلمها وكفاءتها وقدرتها على العمل على أنظمة تشغيل متعددة مثل Windows وmacOS وLinux وUnix [5]. تتيح Python تطوير التطبيقات بسرعة بفضل وجود مكتبة قياسية غنية بتعليمات قابلة لإعادة الاستخدام، مما يقلل الحاجة لكتابة الأكواد من البداية. إضافة إلى أنها تعمل كمفسر ينفذ الأوامر سطراً بسطر، مما يساعد على اكتشاف الأخطاء بسرعة أثناء التنفيذ.

تم استخدام بيئة التطوير المتكاملة PyCharm Community Edition 2023.1.2 المناسبة لتطبيقات Python لتطوير بروتوكول إدارة وتوزيع المفاتيح AinQ الذي تم نمذجته وفق إصدار Python 3.11.3، وقياس بارامتر زمن تنفيذ البروتوكول. أما من أجل اختبار عشوائية مفتاح المجموعة، تم تصميم واجهة تحتوي اختبارات NIST Statistical Test Suite، وذلك باستخدام بيئة التطوير والتعلّم المتكاملة (IDLE) وهي بيئة تطوير متكاملة (IDE) للغة Python مثبتة بشكل افتراضي، واعتماداً على أدوات واجهة المستخدم الرسومية Tkinter.

1. الدراسة المرجعية

قدمت الدراسات [6,7,8] بروتوكولات مركزية لإدارة وتوزيع المفاتيح التي أثبتت قدرتها على ضمان المصادقة والسرية، لكنها عانت من مشكلة نقطة واحدة للفشل إضافة إلى محدودية التوسعية. بالمقابل، توجهت بعض الدراسات [9,10] إلى اقتراح بروتوكولات إدارة مفاتيح لامركزية التي حلت مشكلة نقطة واحدة للفشل وحققت نتائج أكثر فعالية من ناحية السرية الأمامية والخلفية ومقاومة الهجمات، لكنها واجهت أعباء في الحمل الناتج عن حساب تحديثات المفتاح أثناء تغير عضوية المجموعة والذي يساوي $\mathcal{O}$ (n) لمجموعة تضم n عضواً، في حين أن البنية المركزية تتطلب رسالة واحدة فقط من الحجم $\mathcal{O}$ (log n).

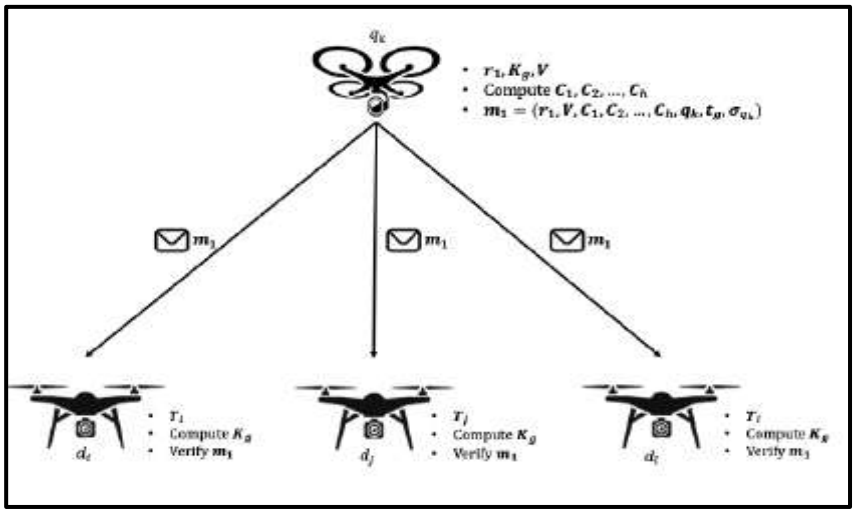
اهتمت العديد من الدراسات بتقديم بروتوكولات تحسن تكاليف توليد وتحديث المفاتيح [11,12,13]، ولكن اقتصرت كفاءتها ضمن بيئات محددة، حيث أن البيئات الديناميكية التي تفرض تغييراً مستمراً في العضوية (عمليات الانضمام والمغادرة المتكررة)، يمكن أن تتحدى أداء هذه البروتوكولات، مما قد يؤدي إلى زيادة التكاليف أو التأخير في توزيع المفاتيح.

تقدم هذه الدراسات جميعها حلولاً مبتكرة ولكنها تواجه تحديات تتعلق بالأمن، والتوسعية، وإدارة المفاتيح في البيئات الديناميكية.

2. البروتوكول Arrows in a Quiver (AinQ) لتوزيع مفاتيح اتصال المجموعة

هو بروتوكول توزيع مفاتيح آمن وخالي من الاقتران ومصادق عليه من المجموعة دون شهادات مصمم خصيصاً لتطبيقات الطائرات دون طيار المتعددة [14]. ينقسم البروتوكول إلى ثلاث مراحل؛ (1) الإعداد والتهيئة، (2) إنشاء المفاتيح وتوزيعها، و(3) تحديث مفاتيح المجموعة عند الانضمام والمغادرة.

قدمت الدراسة التي اقترحت البروتوكول سيناريو لشبكة، كما يبينها الشكل (1)، تتكون من عنقود واحد من الطائرات دون طيار، حيث يقوم قائد العنقود المنتخب q_k في توزيع مفتاح المجموعة K_g على جميع العقد التابعة للعنقود. تفترض الدراسة المذكورة أن جميع الطائرات لديها الحد الأقصى لوقت الطيران وهو t_g ويتم تخزينها في مكان آمن عندما لا تكون في مهمة.



الشكل (1): تطبيق البروتوكول AinQ في شبكة الطائرات دون طيار

تم تقييم أداء البروتوكول من خلال قياس تكلفة الوقت المرتبطة بتوليد مفتاح المجموعة وتحديثه في إجراءات انضمام العضو، ثم انتقل الباحثون إلى إثبات أمن البروتوكول في وجود مهاجم، والذي يمكن أن يكون مهاجماً خارجياً (متنصت سلبي يستمع فقط إلى الشبكة أو كيان خبيث استولى على بعض الطائرات)، أو مهاجماً داخلياً (ممكن أن يكون KGC مخادع أو عضو غادر المجموعة). يضمن تطبيق AinQ حداثة المفتاح حيث يتم اختيار كل مفتاح جلسة جديد عشوائياً، كما يؤمن السرية الماضية عند انضمام عضو جديد. من جانب آخر، فإن اعتماد فريق أو عنقود واحد فقط يمكن أن يؤدي إلى العديد من الآثار السلبية التي يمكن أن تؤثر على الأداء والموثوقية والأمن لهذه الأنظمة، مثل:

- نقطة واحدة للفشل: وجود عنقود واحد يجعل النظام عرضة لانقطاع كامل في حال تعطل هذا العنقود.
- انخفاض الأداء: التحميل الزائد على العنقود الوحيد يؤدي إلى أداء منخفض وزمن استجابة أطول.
- محدودية التوسع: الاعتماد على عنقود واحد يحد من إمكانية التوسع لاستيعاب المزيد من الأعضاء.
- مخاطر أمنية مرتفعة: العنقود الواحد يمثل هدفاً سهلاً للهجمات، وأي اختراق يعرض النظام بأكمله للخطر.

3. اختبارات المعهد الوطني للمعايير والتكنولوجيا National Institute of Standards and Technology (NIST) Statistical Test Suite

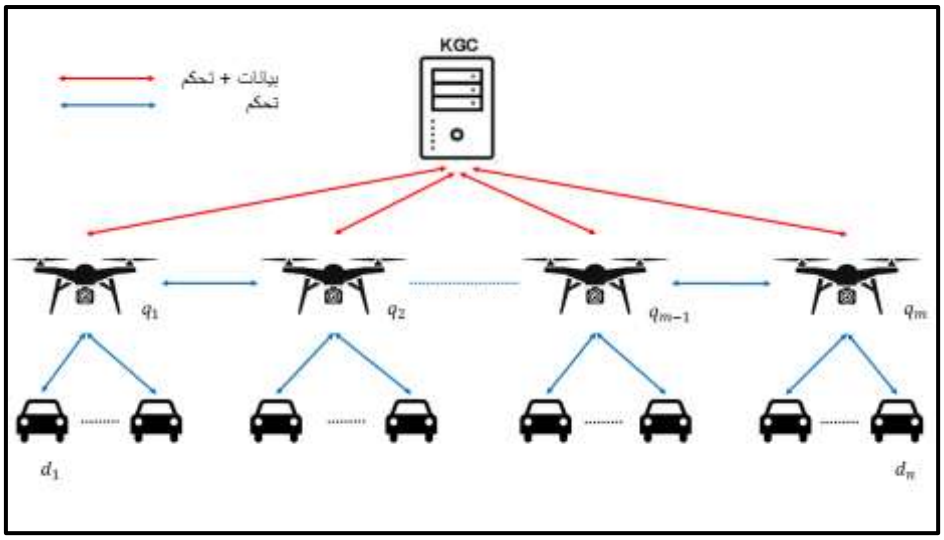
اختبارات NIST لعشوائية المفتاح، والمعروفة باسم NIST Statistical Test Suite، هي عبارة عن حزمة إحصائية تتكون من 15 اختبار تم تطويرها لاختبار عشوائية التسلسلات الثنائية في مفاتيح التشفير المولدة [15]. من خلال تنفيذ هذه الاختبارات بشكل دقيق ومنهجي، يمكن التأكد من أن المفاتيح المولدة تتمتع بمستوى عالٍ من العشوائية ولايمكن التنبؤ بها، مما يعزز أمن البيانات والمعلومات الحساسة.

4. التطبيق المقترح:

4.1. تطبيق البروتوكول Arrows in a Quiver في البنية الموسعة لشبكات العربات بمساعدة الطائرات دون طيار Drones-assisted VANET:

يهدف بحثنا إلى تطبيق البروتوكول Arrows in a Quiver (AinQ) في بنية موسعة من شبكات العربات بمساعدة الطائرات دون طيار تتكون من عدة عناقيد، حيث تمثل الطائرات قادة العناقيد بينما تشكل العربات العقد الأعضاء ضمن العناقيد وفق الشكل (2)، بغية التأكد من أن الشبكة تستطيع التعامل مع زيادة عدد المستخدمين والبيانات بكفاءة

ودون تأثير سلبي على الأداء. إضافة إلى دراسة ديناميكية الشبكة من خلال قياس تكلفة الوقت المرتبطة بتوليد مفتاح المجموعة وتحديثه في إجراءات انضمام مجموعة من الأعضاء ومغادرة آخرين في نفس الوقت.



الشكل (2): تطبيق البروتوكول AinQ في البنية الموسعة لشبكة العربات بمساعدة الطائرات دون طيار

4.2. دراسة عشوائية مفتاح المجموعة للبروتوكول Arrows in a Quiver

تم العمل على تصميم واجهة تتضمن اختبارات NIST لدراسة عشوائية المفاتيح الناتجة عن تطبيق خوارزمية AinQ في النموذج المقترح. تهدف هذه الاختبارات إلى التأكد من أن المفاتيح المولدة تُظهر خصائص عشوائية قوية ولا يمكن التنبؤ بها.

5. البارامترات المدروسة

تم تقييم أداء بروتوكول إدارة المفاتيح المدروس من خلال البارامترات الآتية:

- **زمن تنفيذ البروتوكول:** وهو الزمن الذي يستغرقه بروتوكول إدارة المفاتيح لإتمام العمليات المتعلقة بتوليد مفاتيح جديدة وتوزيعها على أعضاء المجموعة وتحديثها بشكل دوري أو عند حدوث تغير عضوية (انضمام ومغادرة)، مع الأخذ بالحسبان اختلاف الظروف مثل: حجم المجموعة، وتردد حدوث التغييرات، والأحمال الشبكية.
- **عشوائية مفتاح التشفير:** درجة عدم التنبؤ بمفتاح اتصال المجموعة أو استنتاجه من المفاتيح السابقة، ويمكن قياس ذلك من خلال التحقق من عدم وجود أنماط أو تكرارات في سلسلة البتات المكونة للمفتاح، والتأكد من ان القيم المختلفة للمفتاح لا ترتبط ببعضها.

6. إعدادات الدراسة

عند تصميم بنية الشبكة المناسبة لتطبيق البروتوكول "Arrows in a Quiver" على شبكات العربات بمساعدة الطائرات في سيناريوهات عسكرية، ينبغي تحديد عدة عوامل رئيسية تشمل نوع الشبكة، ومعيار الاتصال، ومساحة المكان المدروس، وعدد العقد. كما هو موضح في الجدول (1).

الجدول (1): مواصفات الشبكة المدروسة

نوع الشبكة	MANET (Mobile Ad-hoc Network)
معيار الاتصال	Wi-Fi الذي يعتمد على البروتوكول IEEE802.11

عدد العقد	يصل إلى 1000 عقدة
مساحة المكان المدروس	100 – 500 كم^2

7. سيناريوهات الدراسة

سيتم العمل على عدة سيناريوهات مختلفة تأخذ بالحسبان عدد العناقيد تبعاً لحجم الشبكة، وعدد العربات العقد ضمن كل عنقود، حيث نقوم بدراسة خمس حالات لعدد العقد (50-100-200-500-1000)، إضافة إلى حركة الانضمام والمغادرة للعقد، وذلك بهدف دراسة تأثير كل منها على البارامترات المدروسة وبالتالي كفاءة بروتوكول إدارة المفاتيح.

تشترك جميع السيناريوهات بكون الشبكة مكونة من مركز توزيع مفاتيح KGC ومجموعة من الطائرات دون طيار التي تمثل قادة العناقيد بينما تشكل العربات العقد الأعضاء ضمن العناقيد. تحلق الطائرات على ارتفاعات متوسطة وتعمل كقادة للعناقيد (Cluster Heads). وتمثل العربات الأرضية (مثل مركبات الجنود، والدبابات، أو منصات الأسلحة) العقد (Nodes) في الشبكة. تتصل كل عربة بالطائرة القريبة منها، التي تكون قائد العنقود الخاص بها.

السيناريو الأول: تكون الشبكة عبارة عن عنقود واحد من العربات مع طائرة واحدة تمثل قائد العنقود. تتحرك العقد بشكل مستمر، مما يزيد من تعقيد إدارة المفاتيح.

السيناريو الثاني: الشبكة متوسطة متجانسة مكونة من 20 عنقود، مع نسبة معتدلة من العقد تنضم أو تغادر الشبكة بشكل متواتر. والهدف هو تقييم أداء البروتوكول في شبكة ثابتة نسبياً ومحدودة الحركة.

السيناريو الثالث: الانقطاع المفاجئ لعدة عقد في شبكة من 50 عنقود بسبب خلل تقني أو انقطاع في الاتصال. نهدف في هذا السيناريو إلى اختبار استجابة البروتوكول وإعادة توزيع المفاتيح للعقد المتبقية بشكل سريع وآمن.

السيناريو الرابع: الكثافة العالية مع ديناميكية عالية لشبكة مكونة من 100 عنقود، حيث ينضم ويغادر عدد كبير من العقد باستمرار، وذلك لاختبار قدرة البروتوكول على التكيف مع التغيرات السريعة في الشبكة مع الحفاظ على الاتصال الآمن.

8. التنفيذ العملي

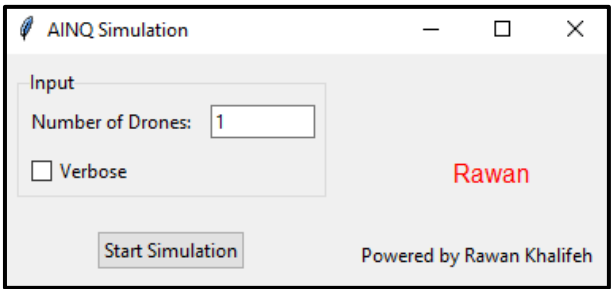
8.1. بيئة المحاكاة

قمنا بكتابة الكود البرمجي المعدل بلغة بايثون الإصدار Python 3.11.3 باستخدام البيئة PyCharm Community Edition 2023.1.2 وتمت المحاكاة على جهاز حاسوب ذو المواصفات المبينة في الجدول (2):

الجدول (2): مواصفات حاسب التنفيذ

نظام التشغيل	المعالج	ذاكرة الوصول العشوائي RAM
Windows 10 Pro	Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz	8 جيغابايت

بعد تنفيذ الكود البرمجي تظهر النافذة المبينة في الشكل (3)، التي تحوي حقلاً لإدخال عدد الطائرات دون طيار التي تمثل قادة العناقيد، ثم بدء المحاكاة Start Simulation. يوفر الخيار verbose عند تفعيله عرض نتيجة تنفيذ كل مرحلة من المراحل السبعة للبروتوكول AinQ بشكل تفصيلي.



الشكل (3): واجهة إدخال عدد العناقيد

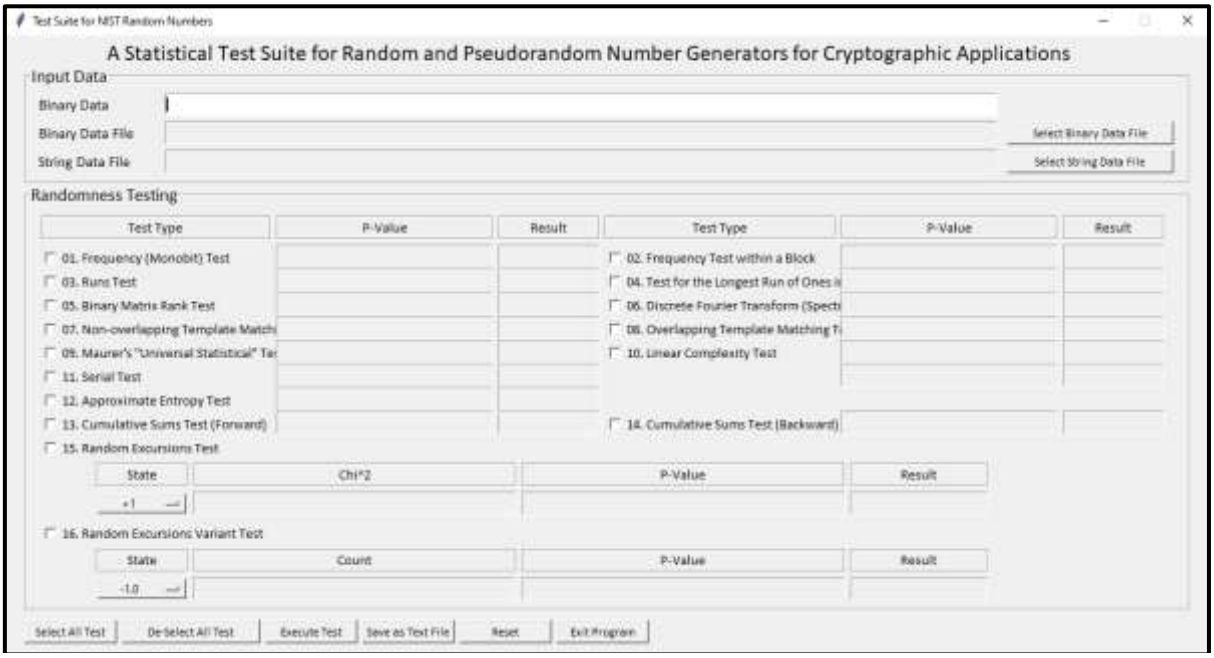
نقوم بعدها بإدخال عدد المركبات في كل عنقود، وعدد المركبات (العقد) المتوقع انضمامها ومغادرتها لعنقودها وفق السيناريو المطبق. بعد ذلك، يبدأ تنفيذ بروتوكول AinQ المعدل لتوليد المفاتيح وتوزيعها وإنشاء مفتاح الجلسة.

8.2. واجهة Test Suite for NIST Random Number

قمنا بتصميم واجهة اختبار مولدات الأرقام العشوائية وشبه العشوائية NIST Statistical Test Suite باستخدام Python 3.11.3، حيث استفدنا من مكتبة Tkinter لتطوير الواجهة الرسومية وفق الخطوات الآتية:

الخطوة 1: إعداد ملفات اختبارات العشوائية: نقوم بتضمين كل اختبار في ملف منفصل يحتوي على دالة تقوم بحساب P-value اعتماداً على مواصفات NIST لكل اختبار.

الخطوة 2: تصميم واجهة المستخدم باستخدام Tkinter: في هذا القسم، قمنا بتصميم واجهة تشمل حقل إدخال المفتاح، ومربعات اختيار لكل اختبار، وأزرار لتشغيل الاختبارات، ونافذة لعرض النتائج كما هو مبين في الشكل (4).



الشكل (4): واجهة اختبارات عشوائية مفتاح المجموعة

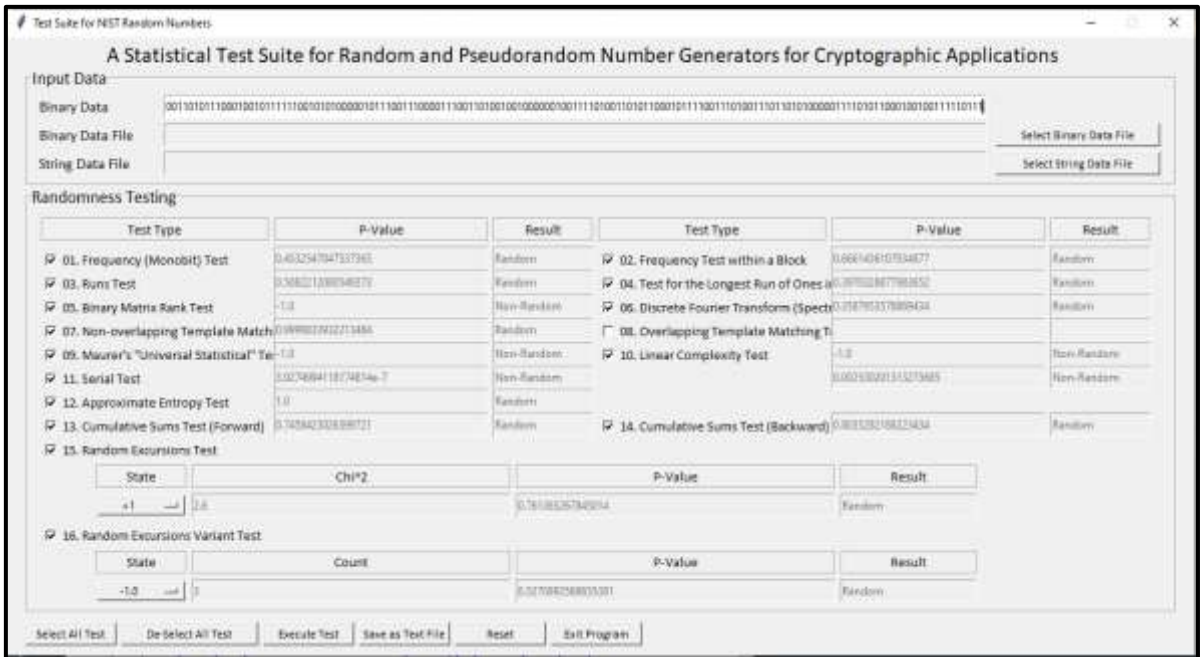
8.3. تخطيط الواجهة

- **النافذة الرئيسية:** تحتوي على العنوان " A Statistical Test For Random And Pseudorandom Number Generators For Cryptographic Applications " وتوفر قائمة بالاختبارات المتاحة، وخيارات التنفيذ المختلفة.

- **إدخال البيانات:** حقل لإدخال سلسلة مفتاح التشفير (أو تحميل ملف نصي يحتوي على المفتاح). يسمح للمستخدم بإدخال سلسة المفتاح المراد اختباره بالصيغة الثنائية. توفر الواجهة إمكانية تحميل ملف نصي يحتوي على المفتاح المراد اختباره في حقل الإدخال إما على شكل محارف، أو بالصيغة الثنائية.
- **قائمة الاختبارات:** قائمة تتضمن كل الاختبارات بحيث يتمكن المستخدم من إجراء اختبار واحد فقط أو مجموعة اختبارات محددة وفق ما يرغب به المستخدم وما يتطلبه التطبيق، كما يمكن تشغيل جميع الاختبارات دفعة واحدة. يضاف إلى جانب كل اختبار حقل القيمة الإحصائية المحسوبة P-value، وحقل تحديد النتيجة (عشوائي أو غير عشوائي).
- **قائمة الأزرار**: بعد إدخال المفتاح واختيار الاختبارات المطلوبة، يمكن الضغط على هذه الخيارات:
  - Execute Test: يتم تنفيذ الاختبار ثم إظهار مربع انتهاء العملية، بعدها إظهار النتائج.
  - Select All Test: عند تفعيله، يتم تحديد جميع مربعات الاختيار الخاصة بالاختبارات الـ15 تلقائياً، بينما يشير De-Select All Tests إلى إلغاء التحديد لجميع الاختبارات.
  - Reset: عند تفعيل هذا الخيار يتم استعادة الإعدادات الأصلية للواجهة وتفريغ جميع الحقول سواء قبل تنفيذ الاختبار أو بعد إظهار النتائج.
  - Save as Text File: يمكن حفظ النتائج في ملف نصي بصيغة (.txt)

8.4. طريقة العمل

- **إدخال المفتاح:** إما بالصيغة الثنائية أو تحميل ملف المفتاح في الحقل المخصص. يمكن اختبار كل المفاتيح التي تولدها جميع العقد في الشبكة والناتجة عن تنفيذ خوارزميات بروتوكول توزيع المفاتيح AinQ، وذلك لقياس مدى عشوائية المفتاح في كل مرحلة من مراحل تنفيذ البروتوكول.
- **اختيار الاختبار**: يقوم المستخدم بتحديد الاختبارات التي يريد تشغيلها.
- **حساب القيم الإحصائية:** يتم حساب القيمة الإحصائية P-value لكل اختبار من اختبارات NIST، ثم تحديد النتيجة من خلال مقارنة قيمة P-value المحسوبة مع العتبة، حيث أن العتبة المختارة عادة هي 0.01 (والتي تعني أن البيانات عشوائية بنسبة 99%) وفق الآتي:
  - إذا كانت P-value > 0.01: يشير إلى أن المفتاح قد اجتاز هذا الاختبار ويعد عشوائياً.
  - إذا كانت P-value <= 0.01: يعني أن المفتاح قد فشل في هذا الاختبار وهناك احتمال عدم عشوائيته.
- **عرض النتائج** : يتم عرض نتائج كل اختبار بالقيمة الإحصائية المحسوبة .P-value، مع تحديد ما إذا كانت هذه القيم تشير إلى عشوائية المفتاح أم لا، كما هو مبين في الشكل (5).



الشكل (5): واجهة عرض نتائج اختبارات عشوائية مفتاح المجموعة

النتائج والمناقشة:

8.5. عرض نتائج تنفيذ البروتوكول AinQ

تُظهر النتائج وفق الشكل (6) زمن توليد القيمة السرية. زمن توليد مفتاح المجموعة Gen Group Key لكل طائرة دون طيار، وزمن استرجاع المفتاح Key Retrieval لكل مركبة ضمن العنقود يوضحه الشكل (7).

```
driver_ainQ
GenSecretValue (Leader Leader 1): 0.049 seconds
GenSecretValue (Leader Leader 2): 0.051 seconds
GenSecretValue (Leader Leader 3): 0.040 seconds
GenSecretValue (Leader Leader 4): 0.047 seconds
GenSecretValue (Leader Leader 5): 0.051 seconds
```

الشكل (6): زمن توليد القيم السرية لقادة العناقيد

```
driver_ainQ
Enter the number of leaved drones for Leader 1: 1
GenSecretValue (Leader Leader 1): 0.017 seconds
GenGroupKey: 0.142 seconds
KeyRetrieval (Vehicle 1): 0.016 seconds
KeyRetrieval (Vehicle 2): 0.015 seconds
KeyRetrieval (Vehicle 3): 0.016 seconds
KeyRetrieval (Vehicle 4): 0.019 seconds
```

الشكل (7): زمن توليد مفتاح المجموعة واستعادته من قبل كل عربة

إضافة إلى زمن تحديث المفتاح عند انضمام مركبات جديدة Re-Key، وزمن تحديث المفتاح عند مغادرة مركبات أخرى، وزمن استعادة المفتاح بعد التحديث لكل مركبة في حال الانضمام وحالة المغادرة كما يوضحه الشكلان (8,9).

```
Group ReKey for Joined Vehicles: 0.102 seconds
KeyRetrieval (vehicle 1): 0.026 seconds
KeyRetrieval (vehicle 2): 0.037 seconds
KeyRetrieval (vehicle 3): 0.025 seconds
KeyRetrieval (vehicle 4): 0.023 seconds
KeyRetrieval (vehicle 5): 0.026 seconds
```

الشكل (8): زمن تحديث المفتاح عند الانضمام واستعادته من قبل العربات

```
Group ReKey for Leaved Vehicles: 0.262 seconds
KeyRetrieval (vehicle 2): 0.051 seconds
KeyRetrieval (vehicle 3): 0.044 seconds
KeyRetrieval (vehicle 4): 0.044 seconds
KeyRetrieval (vehicle 5): 0.045 seconds
```

الشكل (9): زمن تحديث المفتاح عند المغادرة واستعادته من العربات

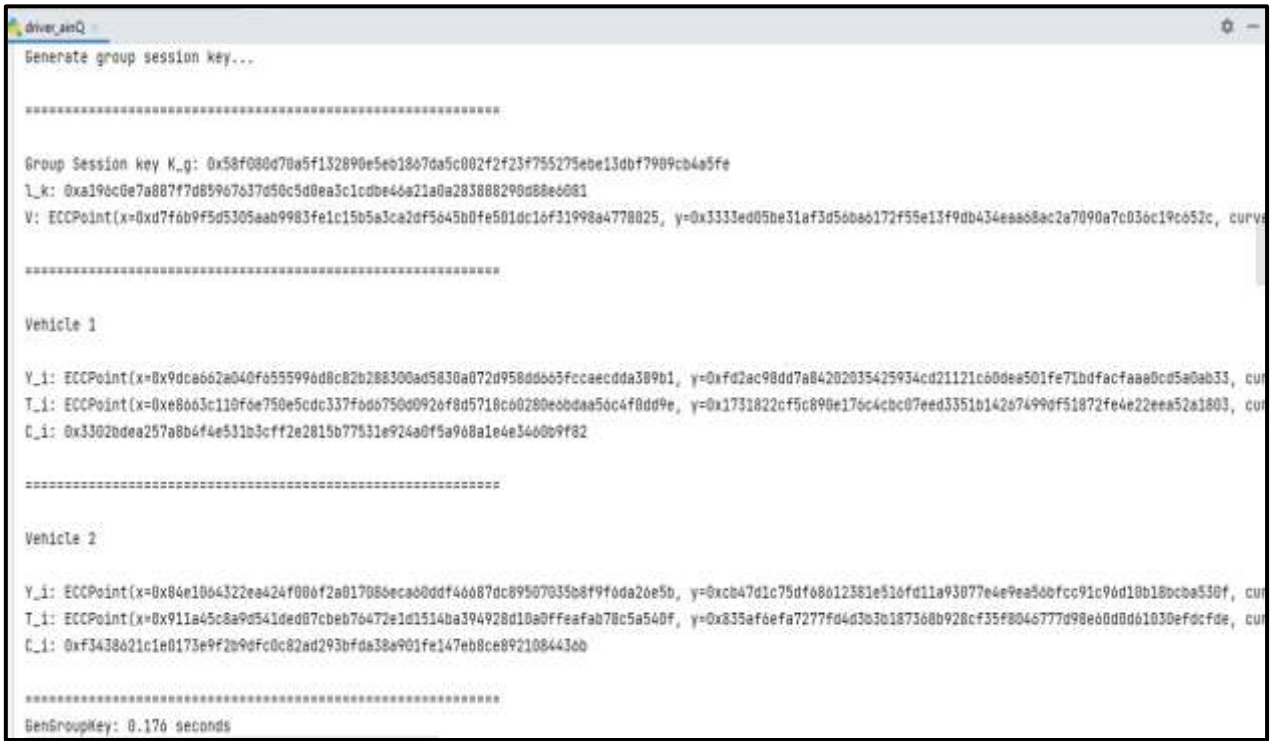
8.6. نتائج تنفيذ البروتوكول مع خيار Verbose:

عند تفعيل خيار verbose يعني ذلك أن عرض النتائج سيكون بشكل تفصيلي على كل مرحلة من مراحل تنفيذ بروتوكول AinQ المعدل، حيث تتضمن مرحلة الإعداد والتهيئة توليد قيمة P على المنحني الإهليلجي واختيار الرقم الاولي q، ثم قيام مركز توليد المفاتيح باختيار القيمة السرية x وحساب المفتاح العام P_{pub}. يليها مرحلة توليد القيم السرية للطائرات قادة العناقيد وتوليد القيم السرية للعربات ضمن كل عنقود كما هو مبين في الشكل (10).

الشكل (10): توليد القيم السرية للطائرات دون طيار والعربات

بعد ذلك، تأتي مرحلة توليد المفاتيح الجزئية العامة والخاصة التي يقوم بتنفيذها مركز توليد المفاتيح KGC لكل قادة العناقيد والعربات ضمن كل عنقود. يظهر الشكل (11) نتائج تنفيذ هذه المرحلة. ثم يبين الشكل (12) نتائج مرحلة توليد مفتاح المجموعة Kg التي يقوم بها قائد كل عنقود، وقائمة النصوص المشفرة التي يرسلها إلى المركبات الأعضاء ضمن عنقوده، يليها مرحلة استعادة المفتاح من قبل كل مركبة كما هو موضح في الشكل (13).

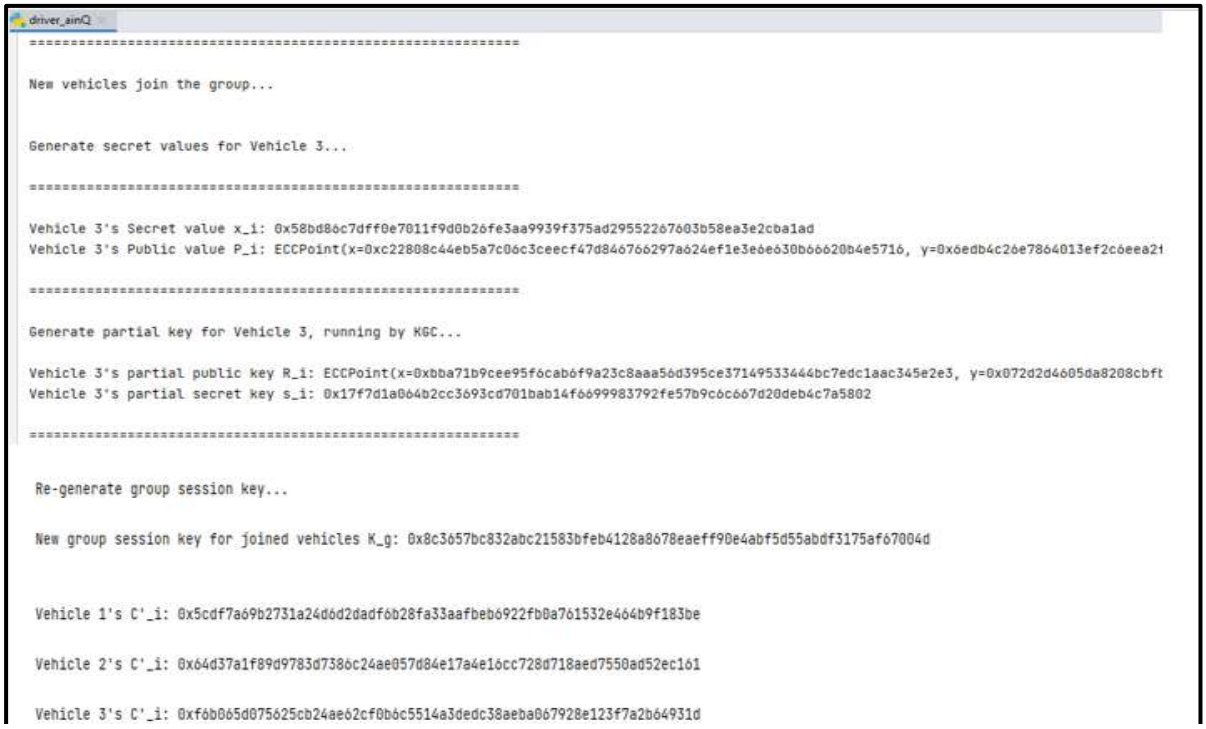
الشكل (11): توليد المفاتيح الجزئية العامة والخاصة للطائرات بدون طيار والعربات



الشكل (12): توليد مفتاح المجموعة وقائمة النصوص المشفرة

الشكل (13): استعادة العربة لمفتاح المجموعة

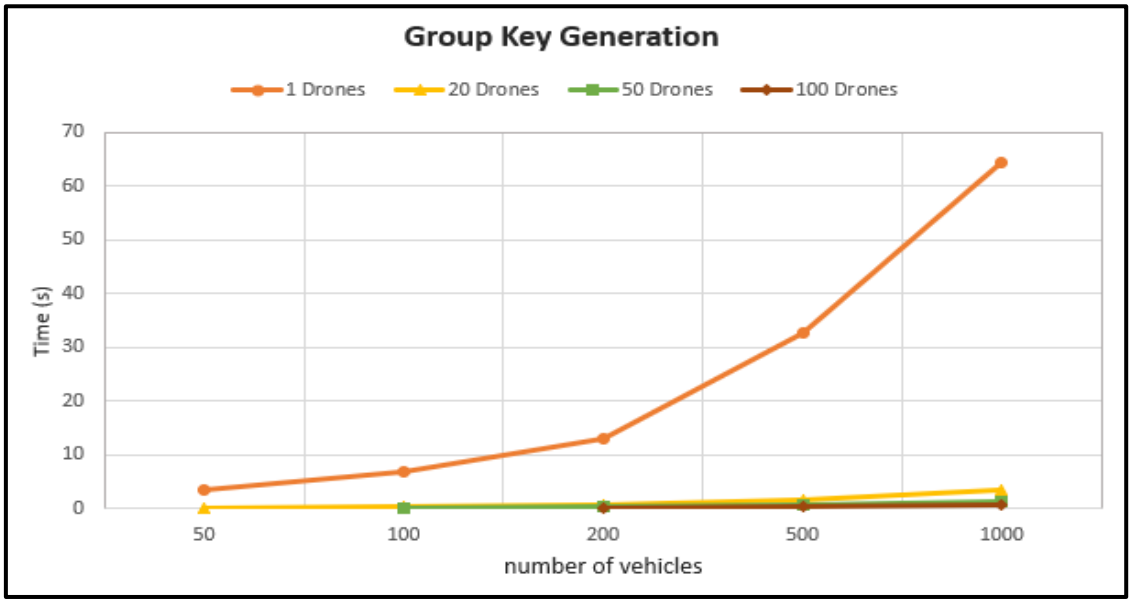
يعرض الشكل (14) نتائج توليد القيم السرية والمفاتيح الجزئية العامة والخاصة الجديدة الناتجة عن انضمام أعضاء جدد، ثم توليد مفتاح الجلسة الجديد. وبنفس الطريقة، تظهر نتيجة تحديث المفاتيح عند مغادرة أعضاء من المجموعة.



الشكل (14): مراحل تحديث مفتاح المجموعة عند الانضمام والمغادرة

8.7. نتائج قياس زمن تنفيذ بروتوكول إدارة وتوزيع المفاتيح AinQ على بنية الشبكة المعدلة

12.3.1 زمن توليد مفتاح المجموعة



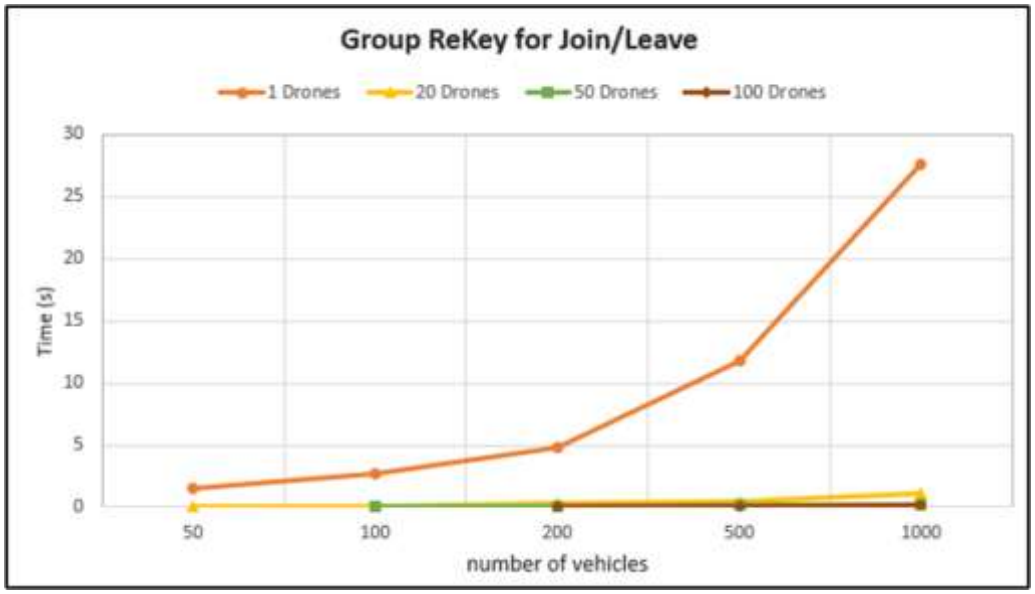
الشكل (15): زمن توليد مفتاح المجموعة لعدد مختلف من العناقيد

يبين الشكل (15) الزمن الذي يستغرقه البروتوكول AinQ لتوليد مفتاح المجموعة (مفتاح الجلسة) للسيناريوهات الأربعة المذكورة في الفصل السابق، عبر أعداد مختلفة من العربات العقد (50 و100 و200 و500 و1000). يُظهر الخط البرتقالي الذي يمثل حالة السيناريو الأول (طائرة دون طيار واحدة)، زيادة كبيرة في الوقت مع زيادة عدد العقد، حيث يرتفع مما يقارب 0 إلى حوالي 70 ثانية مع زيادة العقد إلى 1000. يشير هذا إلى أن زمن توليد مفتاح المجموعة لطائرة دون طيار واحدة يتأثر بشدة بعدد العقد.

بينما تظل الخطوط (الصفراء والخضراء والبنية) التي تعبر عن السيناريوهات الثاني والثالث والرابع على الترتيب، مسطحة نسبياً، حيث نلاحظ أنه تم إنشاء مفتاح المجموعة لشبكة تتكون من 1000 عقدة مقسمة على 20 عنقود بعد 3.4 ثانية، أي أن الانخفاض قد تحسن بنسبة 95%. بالتالي، زيادة عدد العناقيد (20 أو 50 أو 100 طائرة دون طيار) يجعل العملية أقل حساسية لعدد العقد، ويرجع ذلك إلى المعالجة المتوازية وتوزيع الحمل على قادة العناقيد مما يؤدي إلى تنفيذ العمليات بسرعة أكبر.

12.3.2 زمن تحديث مفتاح المجموعة عند الانضمام/ المغادرة

يوضح الشكل (16) تأثير زيادة عدد العناقيد على زمن تحديث مفتاح المجموعة في السيناريوهات الأربعة المختلفة من حيث الديناميكية أي حركات الانضمام والمغادرة للعربات ضمن كل عنقود. نلاحظ أن زمن تحديث المفتاح عند طائرة دون طيار واحدة (الخط البرتقالي) يزداد بشكل كبير مع زيادة عدد العقد، حيث يرتفع من بضع ثوانٍ إلى حوالي 30 ثانية عند الوصول إلى 1000 عربة في الشبكة. يشير ذلك إلى أن الاعتماد على عنقود واحد يعوق الأداء في الشبكات الكبيرة بسبب العبء الزائد.



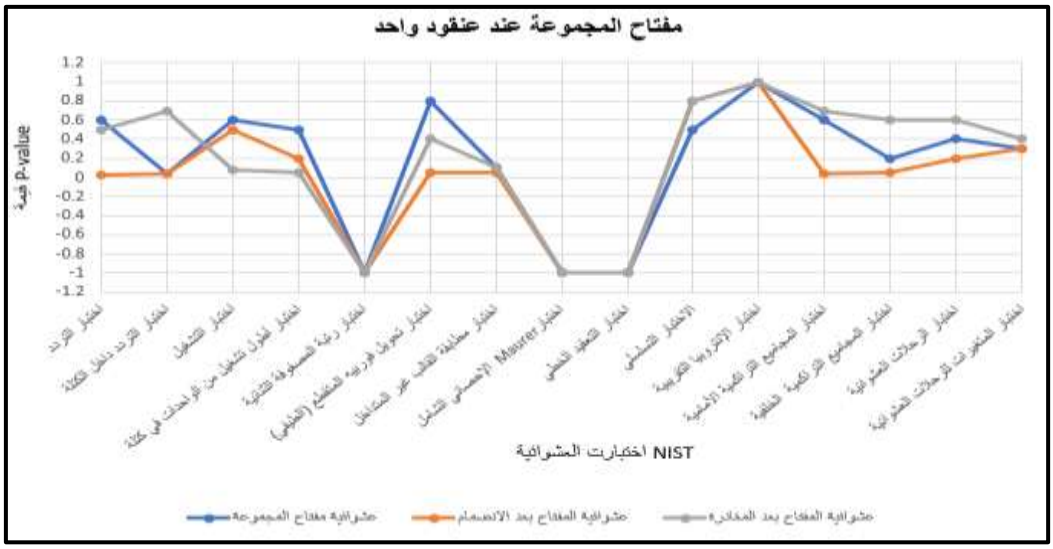
الشكل (16): زمن تحديث مفتاح المجموعة عند الانضمام والمغادرة للسيناريوهات الأربعة

لكن مع زيادة عدد العناقيد أي 20 أو أكثر من الطائرات دون طيار (الخطوط الصفراء، الخضراء، والبنية)، نلاحظ انخفاض زمن تحديث المفاتيح الناتج عن تغير العضوية في المجموعة سواء عند انضمام عقد جديدة أو في حالة مغادرة عقد إلى 0.051 ثانية عند 50 عربة موزعة بالتساوي على 20 عنقود. نستطيع أن نرى أن زمن التحديث يظل ثابتا تقريباً وقريباً من الصفر مما يشير إلى زيادة طفيفة في الزمن مع تزايد العقد تصل إلى ما يقارب 1.15 ثانية عندما يكون عدد العربات الكلي إلى 1000، وذلك بسبب توزيع مرحلة تحديث المفتاح في خوارزمية AinQ على عدة قادة عناقيد أو طائرات، مما يسمح بتنفيذ العمليات بشكل متزامن، وبالتالي ينخفض حجم البيانات المتدفقة عبر كل نقطة في الشبكة. وهو ما يقلل من الزمن المرتبط بإجرائيات تحديث المفتاح وتوزيعه.

12.3.3 نتائج دراسة عشوائية مفتاح المجموعة

تظهر الرسوم البيانية نتائج حساب القيمة P-value لاختبارات مختلفة من NIST لتقييم عشوائية مفاتيح المجموعة في السيناريوهات الأربعة التي تختلف بعدد العناقيد (1،20،50،100)، حيث قمنا باختيار مفاتيح العنقود الأول عندما يكون عدد المركبات 1000 من كل سيناريو. تم اختبار مفتاح المجموعة في ثلاث حالات: "توليد مفتاح المجموعة"، "تحديث المفتاح عند الانضمام"، و"تحديث المفتاح عند المغادرة". ثم مقارنة نتائج الاختبارات بقيمة العتبة 0.01. القيم التي تقل عن العتبة تشير إلى فشل المفتاح في اجتياز اختبار العشوائية، مما يعني وجود نمط معين غير عشوائي.

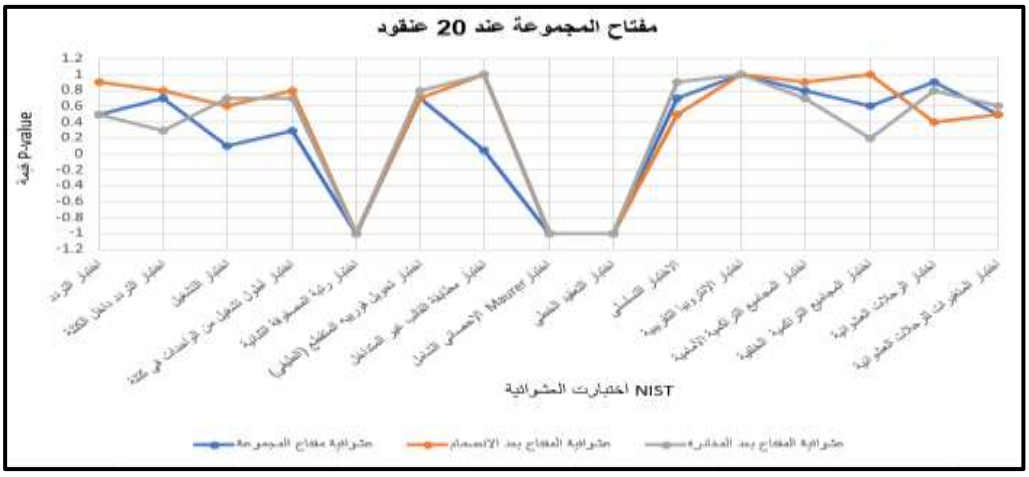
1. تأثير عدد العناقيد على نتائج العشوائية



الشكل (17): عشوائية مفتاح المجموعة لعنقود واحد

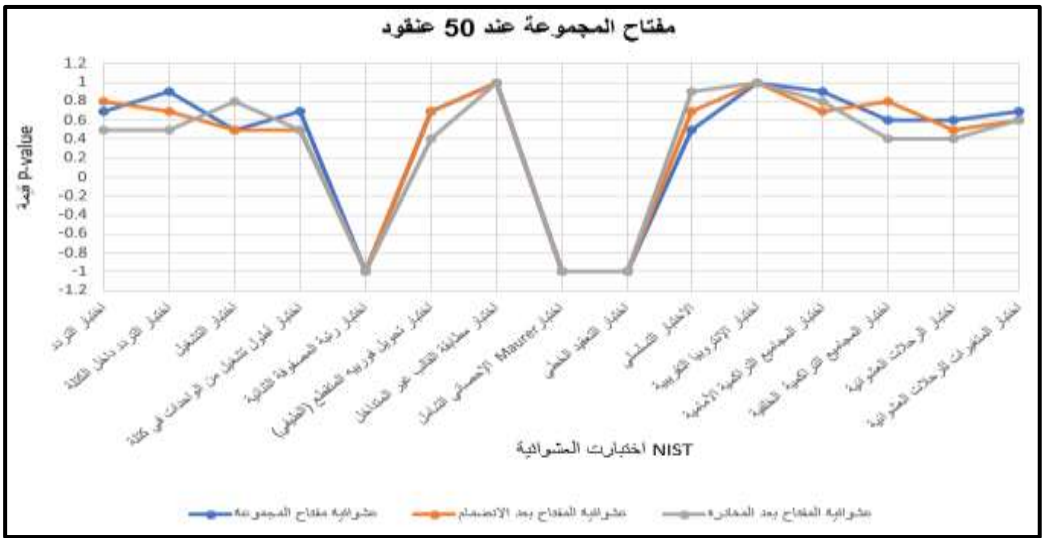
يمثل الشكل (17) سيناريو الشبكة التي تتكون من عنقود واحد. نلاحظ أن النتائج أقل استقراراً وتباين كبير في قيم P-value. بعضها مرتفع نسبياً من 0.5 لتصل في بعض الاختبارات إلى 1 مثل اختبار الانتروبيا التقريبية، مما يشير إلى عشوائية جيدة. بينما في بعض الاختبارات مثل رتبة المصفوفة الثنائية واختبار Maurer الإحصائي الشامل واختبار التعقيد الخطي تظهر قيم أقل من العتبة (تصل إلى −1)، مما يعني أن المفتاح لم يجتز تلك الاختبارات. إضافة إلى ذلك، يجتاز المفتاح اختبار مطابقة القالب المتداخل ولكن بأداء ضعيف (0.05)، مما يعني وجود بعض الأنماط المكررة في المفتاح. يمكننا أن نستنتج أن سيناريو العنقود الواحد يعاني من تكرار الأنماط بسبب محدودية التنوع في توزيع المفتاح، فتكون العشوائية محدودة نسبياً والمفتاح أكثر عرضة لأنماط يمكن التنبؤ بها.

عند سيناريو زيادة عدد العناقيد إلى 20 طائرة دون طيار كما هو موضح في الشكل (18)، نلاحظ تحسناً في قيم P-value مقارنة بالسيناريو الأول، حيث أصبحت أكثر استقراراً حول القيمة 0.5، وتقترب من 1 في اختبارات المجاميع التراكمية الامامية والخلفية واختبار التردد Frequency Test. ولكن ما زالت اختبارات رتبة المصفوفة واختبار Maurer الإحصائي الشامل والتعقيد الخطي تُظهر قيماً منخفضة تجعل المفتاح يفتقر إلى العشوائية الكاملة. نستنتج من ذلك أن زيادة عدد العناقيد يُضيف تنوعًا أكبر في البيانات، مما يُحسن توزيع الأصفار والواحدات. يظهر الشكل (19) تحسن الأداء للسيناريو الثالث عند حالة 50 عنقود أو طائرة دون طيار بشكل ملحوظ، بسبب زيادة التغطية العشوائية والاستقرار في توزيع الأصفار والواحدات مع زيادة عدد العناقيد، حيث يصبح النظام أقل عرضة للأنماط التكرارية مما أدى إلى استقرار في القيم P لمعظم الاختبارات. كذلك الأمر في السيناريو الرابع عند زيادة عدد العناقيد إلى 100 كما هو موضح في الشكل (20).



الشكل (18): اختبارات عشوائية مفتاح المجموعة لـ20 عنقود

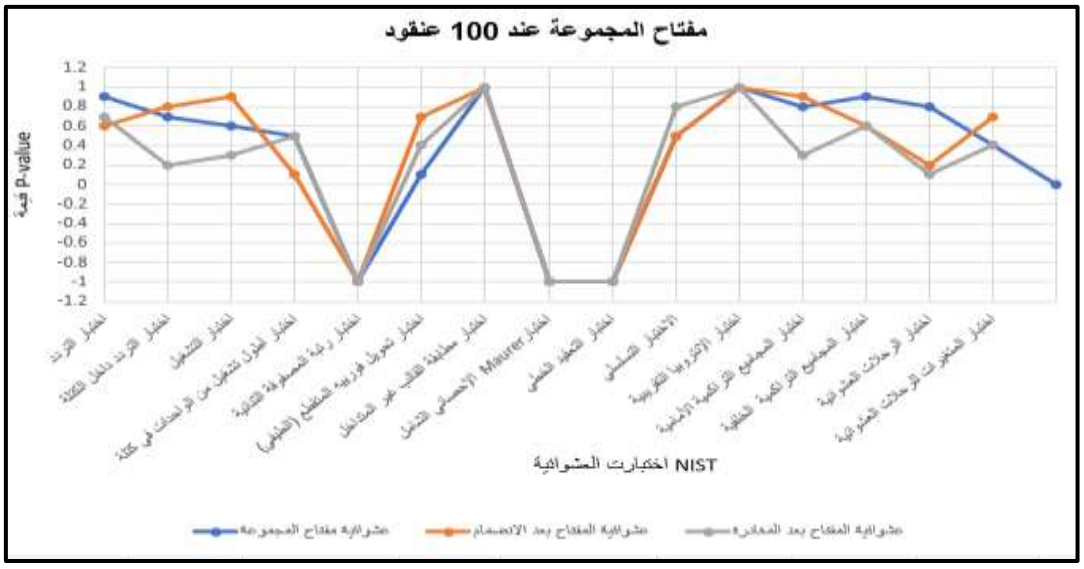
تظهر النتائج استقرار واضح في أداء الاختبارات مقارنة بالسيناريوهات السابقة (خاصة عند أعداد عناقيد أقل)، حيث معظم القيم P-value قريبة من 1. تتراجع القيم السلبية أو القريبة من الصفر باستثناء الأداء المنخفض المشترك لاختبارات رتبة المصفوفة والإحصائي الشامل والتعقيد الخطي في كل السيناريوهات. هذا يعكس أن المفتاح يحقق عشوائية عالية مع زيادة عدد العناقيد حيث يجتاز 12 اختبار من أصل 15.



الشكل (19): اختبارات عشوائية مفتاح المجموعة لـ50 عنقود

2. تأثير ديناميكية الشبكة (حركات الانضمام والمغادرة)

عندما ينضم عضو جديد إلى المجموعة، يساهم في توسيع عدد المفاتيح المستخدمة، مما يحسن من توزيع المفتاح ويقلل احتمال تكرار الأنماط نتيجة اندماج أنماط جديدة من العشوائية التي يجلبها العضو الجديد. أما عند مغادرة عضو حالي من المجموعة، يزال جزء من الأنماط التي كانت موجودة في النظام. إذا كانت المفاتيح المولدة تعتمد على مساهمات العضو المغادر، فإن ذلك قد يؤدي إلى تقليل التنوع وإعادة الأنماط القديمة، وبالتالي عشوائية أقل.



الشكل (20): اختبارات عشوائية مفتاح المجموعة لـ100 عنقود

يمثل الخط البرتقالي في الأشكال السابقة عشوائية مفتاح المجموعة بعد انضمام أعضاء جدد، بينما يمثل الخط الرمادي عشوائية مفتاح المجموعة بعد المغادرة. في حالة السيناريو الأول لعنقود واحد، تؤثر حركات الانضمام والمغادرة بشكل واضح على نتائج بعض الاختبارات بسبب قلة التنوع، مما يجعل الشبكة أكثر عرضة لفقدان العشوائية مع زيادة حجم وديناميكية الشبكة، مما يضعف الأمان. عند عدد قليل من العناقيد (20) كما في السيناريو الثاني، يظهر انضمام مركبات جدد تحسنًا طفيفًا في العشوائية، بينما تسبب مغادرة مركبات انخفاض P-value في بعض الاختبارات، وذلك لأنه في الشبكات ذات المجموعات الصغيرة، تؤدي المغادرة إلى تقليل التنوع، مما يؤثر سلبًا على العشوائية. أما في السيناريوهات الثالث والرابع عند عدد كبير من العناقيد (50 و100)، فإن التنوع الناتج عن انضمام مركبات جدد يقلل من التأثير النسبي لأي نمط متكرر، حيث يساهم في تحسين النتائج عبر جميع الاختبارات تقريبًا. كما يصبح تأثير المغادرة أقل ضررًا مقارنة بالأنظمة الصغيرة وأقل حساسية للتغيرات الناتجة عن الحركات الديناميكية، حيث يتم تعويض نقص الأعضاء بفضل العدد الكبير من العناقيد.

الاستنتاجات والتوصيات:

بعد هذه الدراسة، يمكننا التوصل للاستنتاجات الآتية:

- إن زيادة عدد العناقيد في بروتوكول إدارة وتوزيع المفاتيح Arrows in a Quiver يحسن قابلية التوسع، حيث يحسن بشكل كبير من كفاءة توليد المفاتيح الجماعية في الشبكات الأكبر.
- مع نمو الشبكة إلى 500 أو 1000 عقدة، تواجه طائرة دون طيار واحدة ارتفاعًا حادًا في زمن تحديث مفتاح المجموعة بعد الانضمام والمغادرة، بينما تحافظ 20 طائرة دون طيار أو أكثر على أوقات مستقرة ومنخفضة، مما يجعلها أكثر ملاءمة للشبكات ذات الديناميكية العالية.
- زيادة عدد العناقيد تؤدي إلى تحسن كبير في عشوائية المفتاح الناتج عن تطبيق البروتوكول Arrows in a Quiver. كلما زاد عدد العناقيد، أصبح البروتوكول أكثر قدرة على إنتاج مفاتيح عشوائية تظهر استقرارًا وقوة في

اختبارات NIST . كما أن العناقيد المتعددة تقلل من التأثير السلبي لحركات الانضمام والمغادرة، حيث يكون البروتوكول قادرًا على استيعاب التغيرات مع الحفاظ على عشوائية قوية.

كانت بعض اختبارات العشوائية من معايير NIST ، مثل رتبة المصفوفة الثنائية واختبار Maurer الإحصائي الشامل واختبار التعقيد الخطي، تميل إلى الفشل بشكل متكرر، مما يشير إلى وجود مشكلات في العشوائية. لتحسين هذا، يمكن استخدام مولدات عشوائية أكثر قوة، وتحسين اختيار الأرقام الأولية.

References:

1. Alkubeily, M; Bettahar, H; and Bouabdallah, A. *A New Application-Level Multicast Technique for Stable, Robust and Efficient Overlay Tree Construction*. In Computer Networks (ELSEVIER), 2011.
2. Stachowiak, K; Pawlak, T; and Piechowiak, and M. *Performance Evaluation of Multicast Overlay Routing Protocols*. Image Processing & Communication, 2013.
3. Lin, S., Xu, Y., Wang, H., Gu, J., Liu, J., and Ding, G. *Secure Multicast Communications via RIS Against Eavesdropping and Jamming With Imperfect CSI*. IEEE Transactions on Vehicular Technology, 72, 16805-16810. 2023.
4. Hasan, M.M., Sarkar, M.Z., Rizve, A.Z., and Hasan, M.M. *Enhancing Security in Multicasting Employing Generalized Receiver with SC Diversity and Opportunistic Relaying*. 2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT), 1140-1145. 2024.
5. https://aws.amazon.com/ar/what-is/python/. Last visit: 20/4/2024
6. Taurshia, A., Kathrine, J.W., Andrew, J., and Eunice R, J. *Securing Internet of Things Applications Using Software-Defined Network-Aided Group Key Management with a Modified One-Way Function Tree*. Applied Sciences. 2024.
7. Abdel Hakeem, S, A; Kim, H. *Centralized Threshold Key Generation Protocol Based on Shamir Secret Sharing and HMAC Authentication*. https://www.mdpi.com/journal/sensors. 2022
8. Alkubeily, M. *A New Scheme to Secure Group Communication in Application-Level Multicast Networks.* Tishreen University Journal. Eng. Sciences Series. Vol. 42, N. 5, 2020.
9. Weidner, M; Kleppmann, M; Hugenroth, D; and Beresford, A, R. *Key Agreement for Decentralized Secure Group Messaging with Strong Security Guarantees*. ACM SIGSAC Conference on Computer and Communications Security (CCS '21), 2021.
10. Lian, H., Kang, B., and Yang, L. *Strongly Secure Identity-Based Authenticated Key Agreement Protocol With Identity Concealment for Secure Communication in 5G Network*. IEEE Access, 12, 98611-98622 . 2024
11. Zheng, J., Duan, H., Wang, C., Cao, Q., Xu, G., & Fang, R. *A Drone-Assisted Anonymous Authentication and Key Agreement Protocol with Access Control for Accident Rescue in the Internet of Vehicles*. Electronics. 2024.
12. Luo, H., Wu, Y., Sun, G., Yu, H., & Guizani, M. *ESCM: An Efficient and Secure Communication Mechanism for UAV Networks*. IEEE Transactions on Network and Service Management, 21, 3124-3139. 2023
13. Dammak, M; Senouci, S-M; Messous, M, A; Elhdhili, M, H; Gransart, C. *Decentralized Lightweight Group Key Management for Dynamic Access Control in IoT Environments*. https://hal.science/hal-02965346. 2020.
14. Frimpong, E., Rabbaninejad, R., & Michalas, *A. Arrows in a Quiver: A Secure Certificateless Group Key Distribution Protocol for Drones*. IACR Cryptology ePrint Archive. 2021

15. Bassham, L.E., Rukhin, A.L., Soto, J., Nechvatal, J., Smid, M.E., Barker, E.B., Leigh, S., Levenson, M.S., Vangel, M.G., Banks, D., Heckert, N.A., Dray, J.F., & Vo, S.C. *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*. NIST Special Publication 800-22. 2010