KERANGKA ANALISIS UNTUK PAUTAN RANGKAIAN ANDROID BOT (ANDROID *BOTNET* URLS) DENGAN MENGGUNAKAN TEKNIK ENKRIPSI DAN VISUALISASI

HAIRUL NIZAM BIN BALALO @ BOLALAN

UNIVERSITI KEBANGSAAN MALAYSIA

KERANGKA ANALISIS UNTUK PAUTAN RANGKAIAN ANDROID BOT (ANDROID *BOTNET* URLS) DENGAN MENGGUNAKAN TEKNIK ENKRIPSI DAN VISUALISASI

HAIRUL NIZAM BIN BALALO @ BOLALAN

PROJEK YANG DIKEMUKAKAN UNTUK MEMENUHI SEBAHAGIAN DARIPADA SYARAT MEMPEROLEH IJAZAH SARJANA KESELAMATAN SIBER

FAKULTI TEKNOLOGI DAN SAINS MAKLUMAT
UNIVERSITI KEBANGSAAN MALAYSIA
BANGI

2023

PENGAKUAN

Saya akui karya ini adalah hasil kerja saya sendiri kecuali nukilan dan ringkasan yang tiap-tiap satunya telah saya jelaskan sumbernya.

07 September 2023

HAIRUL NIZAM BIN BALALO @ BOLALAN
P98900

PENGHARGAAN

Dengan nama Allah Yang Maha Pemurah lagi Maha Penyayang.

Segala puji bagi Allah, Tuhan yang Maha Esa, yang dengan limpah kurnia-Nya saya dapat menamatkan penulisan projek ini dengan jayanya walaupun menempuhi pelbagai rintangan. Saya ingin merakamkan penghargaan dan terima kasih yang tidak terhingga kepada semua pihak yang telah memberikan sokongan, bantuan, dan dorongan dalam menyelesaikan projek sarjana ini.

Terima kasih khas ditujukan kepada penyelia projek saya, Dr Wan Fariza Binti Paizi @ Fauzi, terima kasih atas bimbingan, nasihat, dan pandangan yang berharga sepanjang penyelidikan ini.

Kepada kedua ibubapa tersayang, terima kasih atas doa, dan semangat yang tak pernah putus. Mama dan bapa adalah sumber inspirasi dan kekuatan saya. Segala perjuangan dan pengorbanan mama dan bapa selama ini telah memberi saya motivasi yang besar untuk terus berusaha dan mencapai cita-cita. Terima kasih kerana sentiasa berada di sisi saya, memberikan dorongan dan sokongan moral dalam setiap langkah perjalanan ini.

Kepada isteri tercinta, terima kasih atas kesabaran, pengertian, dan kasih sayang yang dicurahkan. Terima kasih kerana selalu ada dalam setiap kesusahan dan kelelahan saya serta sentiasa memberi semangat untuk saya terus melangkah ke hadapan.

Tidak lupa juga kepada adik beradik, keluarga, barisan pensyarah dan semua warga Fakulti Teknologi dan Sains Maklumat (FTSM), Universiti Kebangsaaan Malaysia (UKM) dan sahabat-sahabat di Polis Diraja Malaysia (PDRM) yang memberikan sokongan moral dan motivasi kepada saya.

Akhir sekali, kepada semua individu dan pihak yang telah memberikan sumbangan dalam bentuk apa pun, terima kasih kerana menyumbang kepada kejayaan penulisan projek ini. Semoga segala budi baik yang diberikan mendapat balasan yang berlipat ganda dari Allah.

Semoga projek sarjana ini dapat memberikan manfaat kepada masyarakat dan menjadi asas yang kukuh dalam perjalanan kehidupan ilmiah saya. Terima kasih.

ABSTRAK

Projek ini membentangkan satu rangka kerja analisis untuk *URL botnet* Android dengan menggunakan teknik enkripsi dan visualisasi. Penyelidikan ini mengatasi kelemahan penyelesaian sedia ada dalam mengesan dan menganalisis dengan tepat *URL* yang dienkripsi yang berkaitan dengan aktiviti *botnet* pada peranti Android. Pernyataan masalah menyoroti ketidakcukupan kaedah pengesanan semasa dalam mengenal pasti dengan tepat *URL* yang dienkripsi yang digunakan oleh *botnet* Android. Teknik enkripsi yang digunakan oleh penulis *malware* menjadikan sukar untuk menerjemahkan niat sebenar *URL* ini, menghalang analisis dan pengesanan yang efektif terhadap aktiviti *botnet*. Untuk mengatasi cabaran ini, satu rangka kerja analisis dicadangkan. Rangka kerja ini menggunakan teknik enkripsi dan kod *Python* untuk mendekripsi dan menerjemahkan *URL* yang berkaitan dengan tiga keluarga perisian berbahaya yang berbeza: *Anserverbot*, *PJapps*, dan *Droiddream*. Dengan mendekripsi dan menterjemahkan *URL*, sifat sebenar aktiviti *botnet* dapat didedahkan, membantu dalam memahami dan mengesan dengan lebih baik. Projek ini melaporkan keputusan yang berjaya dicapai melalui pelaksanaan rangka kerja analisis. Kod *Python* yang dibangunkan berjaya mendekripsi dan menerjemahkan *URL* keluarga perisian berbahaya yang telah disebutkan, membolehkan pemahaman yang lebih mendalam tentang tingkah laku dan fungsi mereka. *URL* yang telah didekripsi kemudiannya divisualisasikan menggunakan alat-alat visualisasi, membolehkan representasi yang lebih intuitif tentang aktiviti *botnet*. Hasil kajian ini memberikan sumbangan kepada bidang analisis *botnet* Android dengan menyediakan rangka kerja yang komprehensif yang berkesan dalam mengendalikan *URL* yang dienkripsi. Keupayaan untuk mendekripsi dan menerjemahkan *URL* meningkatkan kecekapan dan kebolehpercayaan dalam pengesanan dan analisis *botnet*, membantu dalam mitigasi ancaman yang berkaitan dengan *botnet* pada peranti Android. Kajian ini menunjukkan potensi teknik enkripsi dan visualisasi dalam menganalisis *URL botnet* Android, serta penggunaan praktikal melalui rangka kerja yang dicadangkan. Penyelidikan akan datang boleh terus meneroka dan memperluas rangka kerja ini, termasuk keluarga perisian berbahaya tambahan dan penyempurnaan teknik visualisasi untuk meningkatkan pemahaman tentang tingkah laku *botnet*.

ANALYSIS FRAMEWORK FOR ANDROID BOTNET URLS USING ENCRYPTION TECHNIQUES AND VISUALIZATIONS

ABSTRACT

This project presents an analysis framework for Android botnet URLs by leveraging encryption techniques and visualizations. The research addresses the limitations of existing solutions in accurately detecting and analyzing encrypted URLs associated with botnet activities on Android devices. The problem statement highlights the inadequacy of current detection methods in accurately identifying encrypted URLs used by Android botnets. Encryption techniques employed by malicious actors make it challenging to decipher the true intent of these URLs, hindering effective analysis and detection of botnet activities. To overcome these challenges, an analysis framework is proposed. The framework utilizes encryption techniques and Python code to decrypt and decode the URLs associated with three distinct malware families: Anserverbot, PJapps, and Droiddream. By decrypting and decoding the URLs, the true nature of the botnet activities can be revealed, aiding in better understanding and detection. The project reports successful results achieved through the implementation of the analysis framework. The Python code developed managed to decrypt and decode the URLs of the aforementioned malware families, enabling a deeper insight into their behavior and functionality. The decrypted URLs were then visualized using visualization tools, allowing for a more intuitive representation of the botnet activities. The outcomes of this research contribute to the field of Android botnet analysis by providing a comprehensive framework that effectively handles encrypted URLs. The ability to decrypt and decode URLs enhances the accuracy and reliability of botnet detection and analysis, thereby aiding in the mitigation of botnet-related threats on Android devices. This study demonstrates the potential of encryption techniques and visualizations in analyzing Android botnet URLs, and their practical application through the proposed framework. Future research can further explore and expand upon this framework, incorporating additional malware families and refining the visualization techniques to enhance the understanding of botnet behaviors.

KANDUNGAN

Halaman

SENARAI JADUAL

SENARAI ILUSTRASI

SENARAI SINGKATAN

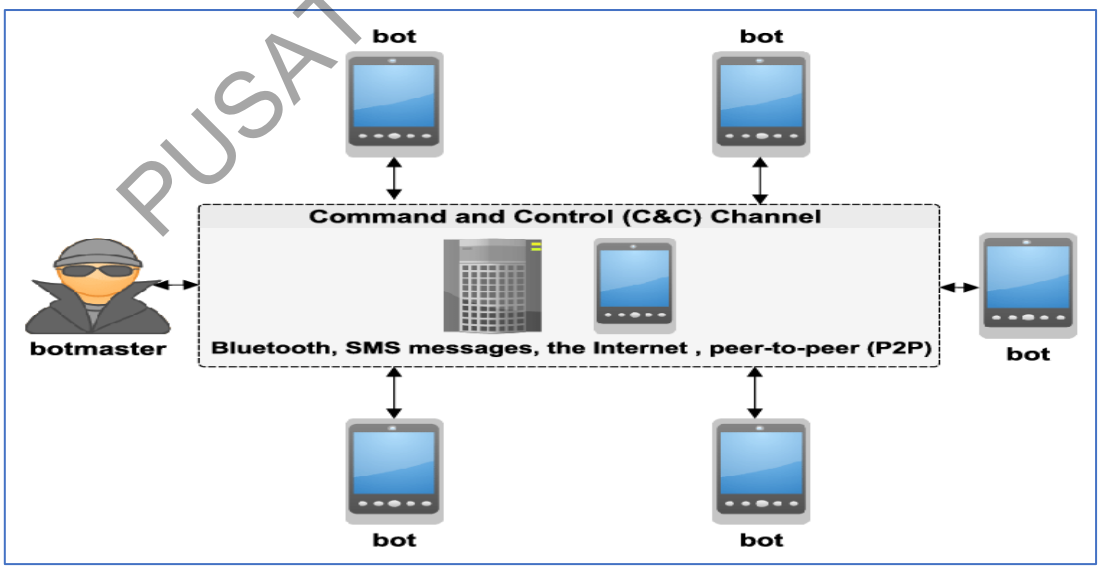
AES	Advanced Encryption System
C&C	Command and Control
DoS	Denial of Service
DDoS	Distributed Denial of Service
P2P	Peer-to-Peer
SMS	Short Messaging Service
UKM	Universiti Kebangsaan Malaysia
URL	Uniform Resource Locator

BAB I

PENGENALAN

1.1 PENDAHULUAN

Keselamatan rangkaian komputer dan pengurusan risikonya merupakan cabaran penting dalam dunia siber. Kini, serangan siber semakin bertambah maju dan kompleks, mengakibatkan kesan yang merugikan kepada kebanyakan pengguna dan organisasi. Dalam usaha untuk memerangi serangan siber, pengesanan rangkaian bot (*botnet*) telah menjadi fokus utama dalam kajian keselamatan siber pada masa ini.

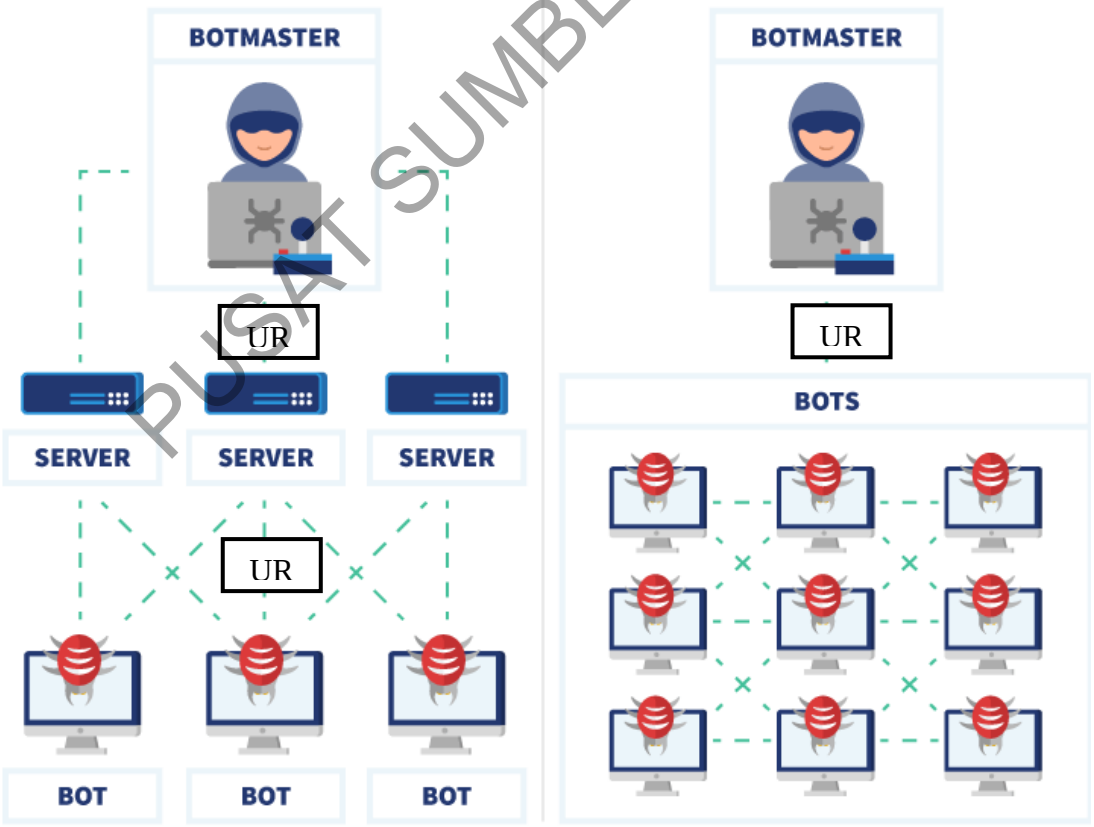


Rajah 1.1 Komponen rangkaian bot pada peranti pintar

Sumber: Alqatawna & Faris 2017

Botnet adalah rangkaian komputer yang dikawal secara sentral dan digunakan untuk tujuan jahat seperti penghantaran spam, pancingan data (*Phishing)*, penggodaman perbankan dalam talian, dan serangan keselamatan *Denial of Service* (*DoS*) atau *Distributed Denial of Service* (DDoS). **Rajah 1.1** menunjukkan komponen *botnet* pada peranti pintar yang mangandungi *botmaster* dan bot. Penjahat internet menggunakan *Command and Control (C&C) channel* seperti *Bluetooth, Short Messaging Service (SMS), Internet*, dan *Peer-to-Peer (P2P)* untutk menyebarkan aktiviti jahat mereka (Alqatawna & Faris 2017).

URL (*Uniform Resource Locator*) memainkan peranan penting dalam operasi *botnet*. *URL* digunakan oleh pengendali *botnet* (*botmaster)* untuk menghubungkan peranti terinfeksi (*zombies*) dengan server yang mengendalikan *botnet* (*Command and Control Server*) mengikut jenis-jenis *botnet* (*centralized botnet* dan *decentralized botnet*) seperti yang ditunjukkan di **Rajah 1.2** dan **Rajah 1.3**.



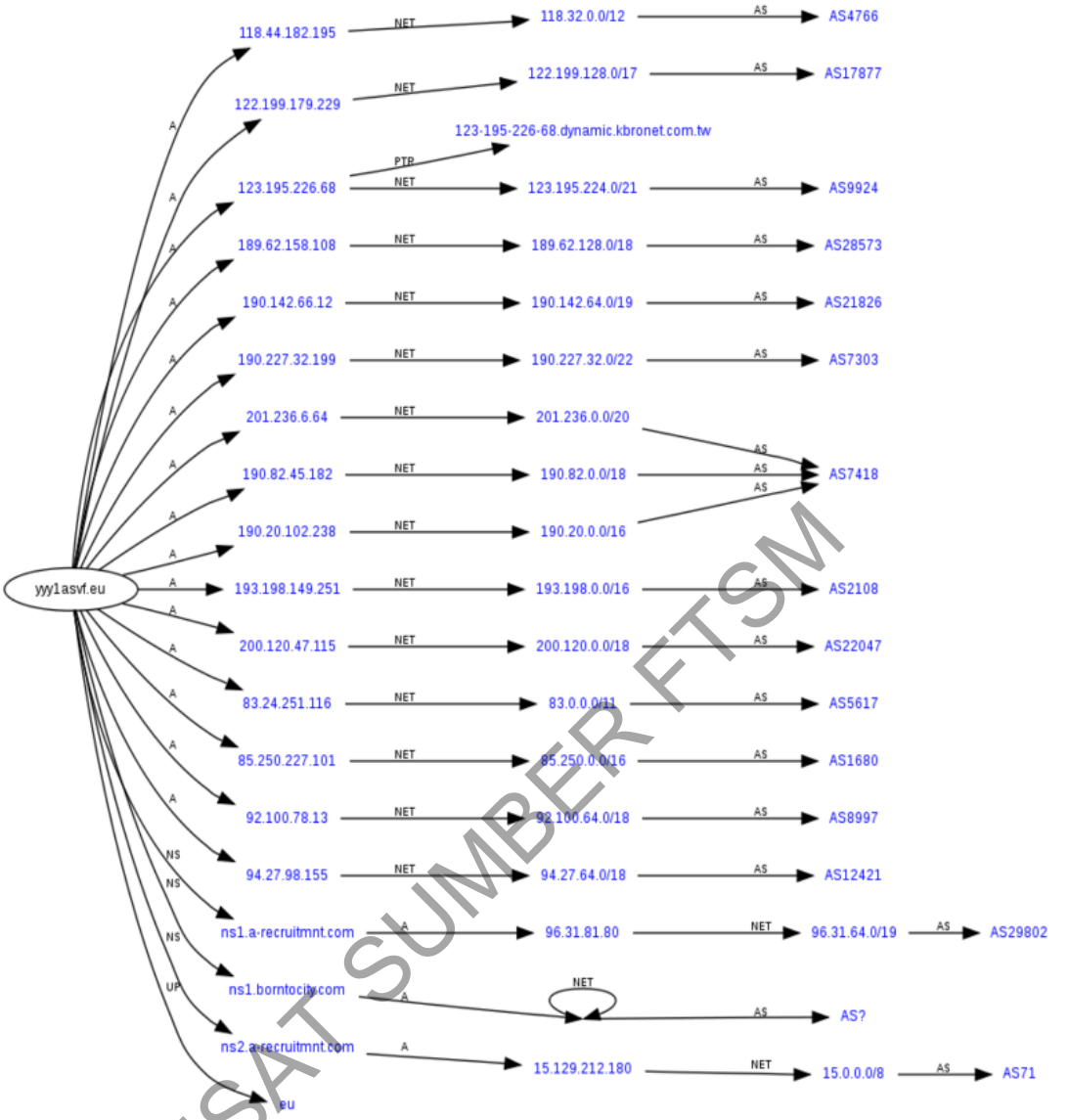
Rajah 1.2 Jenis-jenis *Botnet*: *centralized botnet* dan *decentralized botnet*

Sumber: Marks 2023

URL ini berfungsi sebagai alamat yang digunakan oleh *botnet* untuk berkomunikasi dengan server pusat (*botmaster*) atau sumber daya lain (*fast-flux*) yang diperlukan untuk melaksanakan aktiviti jahat.

Seperti yang ditunjukkan di **Rajah 1.2**, pengendali *botnet* akan menghantar arahan kepada *botnet* melalui *URL* yang disertakan dalam *malware* yang terpasang di peranti terinfeksi. *URL* juga digunakan untuk menghantar data yang diperlukan untuk operasi *botnet*, seperti mengumpulkan maklumat mangsa atau menghantar hasil aktiviti jahat melalui daya lain dengan menggunakan dua jenis *botnet* iaitu *botnet* berpusat (*centralized*) dan *botnet* tidat berpusat (*decentralized*). *Botnet* berpusat mengandalikan satu titik pusat yang dikenali sebagai C&C server untuk mengirimkan arahan kepada bot yang dijangkiti. Namun, jika C&C server dikesan dan dinyahaktifkan, *botnet* berpusat dapat terhenti. *Botnet* tidak berpusat pula tidak bergantung pada satu titik pusat. Pengendalian dan pengawasan bot tersebar di beberapa entiti, menjadikannya lebih sukar untuk dihentikan. Walaupun salah satu entiti pengendalian dihentikan, *botnet* tidak berpusat masih dapat beroperasi melalui entiti pengendalian yang lain. Oleh itu, *botnet* tidak berpusat lebih kukuh dan sukar untuk dikesan.

Merujuk kepada **Rajah 1.3**, *fast-flux botnet* yang digunakan mempunyai banyak *IP addresses* yang didaftar di bawah domain yang sama (1 domain boleh memiliki hingga 15 *IP addresse*s). *Botnet fast-flux* ini menggunakan pendekatan yang dinamik dan kompleks untuk menyembunyikan server perintah dan kawalan (C&C). Dalam *botnet fast-flux*, C&C server tidak berada di lokasi yang tetap. Sebaliknya, *botnet* ini menggunakan jaringan yang berubah-ubah dengan cepat untuk menyembunyikan identiti sebenar C&C server. Tujuan utama *botnet fast-flux* adalah untuk menyembunyikan infrastruktur mereka dan menyukarkan usaha untuk memadamkan *botnet*. *Botnet fast-flux* sering digunakan untuk aktiviti jahat seperti penyebaran *malware*, serangan DDoS, dan penipuan dalam talian. *Botnet* ini membolehkan pelaku jenayah siber untuk menjalankan serangan secara bersembunyi, sambil mempertahankan infrastruktur mereka dengan cara yang sangat dinamik (Wikiwand - Fast Flux n.d).



Rajah 1.3 Contoh penggunaan *URL* dan IP address untuk *botnet* Fast-Flux

Sumber: Wikiwand - Fast Flux n.d

Mengikut kajian yang dilakukan oleh Xu et al. (2014), penggunaan enkripsi dalam *botnet* semakin meningkatkan kesukaran dalam pengesanan, terutamanya apabila *botnet* tersebut menggunakan *URL* yang dienkripsi dan telah diubah suai mengikut jenis ataupun varian *botnet*. Teknik yang canggih ini mengakibatkan kelewatan dalam memerangi serangan siber. Untuk mengatasi masalah ini, kerangka analisis untuk pautan rangkaian bot yang dienkripsi akan dibina bagi memerangi serangan siber yang dihasilkan oleh *botnet*.

Dalam projek ini, akan dijelaskan konsep *botnet*, jenis ataupun varian *botnet* serta penggunaan enkripsi dalam *botnet* yang digunakan untuk mengelakkan pengesanan. Kajian ini juga akan menerangkan tentang teknologi yang digunakan dalam mengesan dan menganalisa *botnet*, terutamanya *botnet* yang menggunakan enkripsi dalam komunikasi melalui pautan *URL*.

1.2 PERMASALAHAN KAJIAN

Botnet telah menjadi ancaman besar terhadap keselamatan internet selama beberapa tahun. Jenayah yang dilakukan menggunakan *botnet* telah meningkat dari 2,227 kes pada tahun 2019 kepada 4,497 kes pada tahun 2020, menunjukkan peningkatan sebanyak 102%. Jenayah siber yang melibatkan penggunaan *botnet* turut memberi kesan besar terhadap sektor perbankan, dengan 63.5% kes jenayah *botnet* berkaitan dengan penipuan perbankan dalam talian. Pada tahun 2022, jumlah ini meningkat kepada 129 134 kes (CyberSecurity Malaysia 2023).

Hingga kini, *botnet* semakin menjadi ancaman yang semakin besar terutamanya bagi pengguna peranti Android. Hal ini terlihat dari peningkatan jumlah serangan *botnet* Android yang terus meningkat dari tahun ke tahun, sebagaimana tercatat dalam laporan statistik terbaru yang diterbitkan oleh McAfee pada tahun 2021 (McAfee 2021). Pengesanan *botnet* bagi peranti Android yang melibatkan penggunaan enkripsi sebagai teknik serangan telah menjadi fokus utama para pengkaji dari bidang akademik mahupun industri sejak kemunculan *botnet* Android yang pertama pada March 2011, (Bilge et al. 2011).

Namun begitu, evolusi *botnet* yang mempunyai banyak varian telah membuatkan pengesanan menjadi rumit dan sulit. Kini, *botnet* menggunakan pelayan *Command & Control* (C&C) yang dienkripsi pula untuk berkomunikasi dengan peranti yang terkompromi sebagai medium komunikasi dengan *botnet* mereka. *Botnet* yang menggunakan enkripsi pada *URL* semakin meningkatkan kesukaran dalam pengesanan, terutamanya apabila enkripsi yang digunakan itu telah diubahsuai secara unik mengikut teknik dan varian tertentu.

Kajian sedia ada mengenai pengesanan *botnet* Android sering memberi tumpuan kepada pelayan C&C tradisional, tetapi gagal menangani penggunaan C&C yang moden dan canggih (pengubahsuaian teknik enkripsi mengikut varian), sekaligus mencipta gap dalam keselamatan peranti mudah alih. Hal ini menyebabkan kajian dan produk pengesanan *botnet* yang sedia ada tidak mampu mengesan pengunaan enkripsi secara tepat, seperti yang dijelaskan dalam laporan yang diterbitkan oleh Symantec pada tahun 2018 (Symantec 2018) dan Spamhaus pada tahun 2022 (Spamhaus 2022). Hal ini telah dibuktikan melalui laporan *VirusTotal* (*VirusTotal* 2023) Produk-produk yang seperti AlienVault, BitDefender, Avira, ADMINUSLabs, dan Baidu-International.

Penggunaan teknik enkripsi dalam komunikasi *botnet* Android menyajikan cabaran dalam mengesan dan menganalisis lalu lintas *botnet*, menekankan keperluan untuk mempunyai kerangka analisis dan pengesanan yang kukuh, fleksibel, dan boleh dikembangkan. Kerangka-kerangka yang sedia ada memfokuskan teknik pengesanan dan enkripsi *URL* secara umum dan tidak mengambil kira teknik enkripsi yang unik mengikut varian khas. Contohnya, *heuristic and behavioral analysis framework* seperti *FireEye Malware Analysis*[1], dan *Cuckoo Sandbox*[2] hanya mengamati perilaku *URL botnet* yang dienkripsi dalam lingkungan yang dikawal untuk mengenal pasti corak dan tindakan yang mencurigakan. Kerangka lain yang dikenali dengan nama *cipher and decryption analysis framework* pula menganalisis teknik enkripsi yang digunakan dalam *URL botnet* dan mencuba mendekripsi *URL* tersebut dengan memahami algoritma dan kunci enkripsi yang mungkin digunakan.

Ketiadaan kerangka analisis yang kukuh dan fleksibel untuk menganalisa *URL botnet* yang dienkripsi mengikut varian mencipta risiko yang besar bagi perniagaan dan individu, yang mungkin mengalami kerugian kewangan atau kecurian data akibat aktiviti *botnet*. Oleh itu, terdapat keperluan untuk membangunkan teknologi pengesanan yang baru dan lebih berkesan bagi mengesan dan menganalisa *botnet* yang menggunakan enkripsi untuk berkomunikasi.

---

[1] https://www.threatprotectworks.com/FireEye-AX-Series.asp

[2] https://cuckoosandbox.org/fireeye

1.3 OBJEKTIF KAJIAN

Berdasarkan permasalahan yang dinyatakan di **Bahagian 1.2**, berikut adalah objektif yang ingin dicapai dalam kajian ini:

1. Mengkaji sistem dan rangka kerja terkini untuk mengesan *Android Botnet URLs* yang dienkripsi dan diubahsuai mengikut jenis varian.

2. Mengenal pasti komponen dan modul yang terlibat dalam proses menyahkod dan menyahsulit *Android Botnet URLs*.

3. Merekabentuk dan membangunkan kerangka analisis dan visualisasi untuk *Android Botnet URLs* yang dienkripsi secara unik mengikut jenis varian.

1.4 SKOP KAJIAN

Terdapat beberapa skop kajian yang perlu dijelaskan:

1. Kajian ini tidak merangkumi pengesanan atau pengelasan Android yang menunjukkan perbezaan *botnet* Android (*malware)* dari aplikasi yang normal (*benign)*.

2. Kerangka ini hanya menganalisis dan mengenal pasti *Android Botnet URLs* di dalam *botnet* varian yang menggunakan enkripsi pada. Oleh itu analisis perbezaan antara *malware* dan *benign* bukan di dalam skop kajian ini.

3. Hanya tiga (3) varian *botnet* (*Anserverbot*, *PJapps*, dan *Droiddream*) yang digunakan untuk menguji kerangka analisis yang dicadangkan di dalam kajian ini. Pilihan untuk menggunakan hanya tiga varian *botnet* adalah berdasarkan beberapa faktor seperti berikut:

    a. Kepentingan Keselamatan: Ketiga-tiga varian *botnet* yang dipilih merupakan ancaman yang signifikan atau berkaitan dengan serangan

yang telah dilaporkan secara meluas. Menguji kerangka analisis pada varian *botnet* yang relevan dapat memberikan maklumat yang berharga tentang keberkesanan kerangka tersebut dalam menghadapi ancaman di dalam siber.

b. Keterbatasan Sumber Daya: Menguji kerangka analisis pada beberapa varian *botnet* memerlukan sumber daya dan masa yang mencukupi. Keterbatasan sumber daya menjadi faktor dalam memilih hanya tiga varian *botnet* yang mewakili untuk diuji.

c. Keterbatasan Data: Untuk menguji kerangka analisis, data yang sah dan boleh dipercayai tentang tingkah laku varian *botnet* yang dipilih diperlukan.

Walaupun hanya tiga varian *botnet* yang digunakan, hasil ujian pada varian tersebut dapat memberikan pemahaman yang berharga tentang keupayaan dan keberkesanan kerangka analisis yang dicadangkan dalam menghadapi serangan *botnet* yang serupa atau berasal dari varian serangan yang sama.

1.5 STRUKTUR LAPORAN

Laporan ini terdiri daripada enam bab: pengenalan, kajian literatur, kaedah kajian, pelaksanaan kajian, dapatan kajian, dan rumusan dan cadangan:

1. **Bab pertama:** Pengenalan, merupakan pendahuluan untuk memperkenalkan topik laporan, diikuti dengan penerangan mengenai permasalahan kajian yang akan dikaji. Objektif kajian juga dinyatakan sebagai matlamat utama laporan ini, dan skop kajian diberikan untuk memberikan batasan terhadap isu yang akan dibincangkan.

2. **Bab kedua:** Kajian Literatur, memperkenalkan topik secara lebih mendalam dengan memberikan penjelasan mengenai varian *Botnet* untuk Android. Selain itu, tinjauan literatur akademik dan industri disediakan untuk memberikan

landasan teori dan praktikal yang berkaitan dengan topik laporan ini. Perbandingan sistem sedia ada juga dilakukan untuk melihat kelebihan dan kekurangan sistem yang telah ada.

3. **Bab ketiga:** Kaedah Kajian menjelaskan mengenai metodologi yang digunakan dalam kajian ini. Reka bentuk kerangka kajian juga diperkenalkan untuk memberikan panduan dalam melaksanakan kajian.

4. **Bab keempat:** Pelaksanaan Kajian menerangkan tentang pelaksanaan kajian, termasuk pengumpulan dan pemprosesan data. Pembangunan kerangka analisis dan pengaturcaraan algoritma juga dibincangkan di sini.

5. **Bab kelima:** Dapatan Kajian membincangkan dapatan kajian yang diperoleh berdasarkan analisis penggunaan enkripsi dan penggunaan visualisasi yang telah dilakukan.

6. **Bab keenam:** Rumusan dan Cadangan merangkumkan keseluruhan kajian dengan memberikan rumusan terhadap dapatan yang diperoleh. Selain itu, beberapa cadangan juga diberikan untuk tindakan masa depan dalam bidang kajian ini.

7. **Rujukan** disediakan pada akhir laporan untuk memberikan sumber rujukan bagi bahan-bahan yang digunakan dalam laporan ini. Lampiran-lampiran yang terdiri daripada skrip dan senarai sampel mengikut varian *botnet* juga disertakan bagi memberikan maklumat tambahan yang berkaitan.

BAB II

KAJIAN LITERATUR

2.1 PENGENALAN

Bab ini membincangkan hasil kajian terdahulu berkaitan dengan *botnet* untuk peranti pintar Android, jenis-jenis varian *botnet*, dan juga kepelbagaian modus operandi (tingkah laku). Tinjauan literatur dari bidang akademik dan industri serta perbandingan sistem sedia ada juga akan dibincangkan dengan terperinci. Berdasarkan keseluruhan maklumat yang dikumpul, kerangka analisis untuk *Android Botnet URLs* dengan menggunakan teknik enkripsi dan visualisasi akan diusulkan di hujung bab ini.

2.2 VARIAN *BOTNET* UNTUK ANDROID

Varian *botnet* untuk Android merujuk kepada rangkaian *botnet* yang disasarkan untuk peranti Android untuk digunakan dalam aktiviti jahat tanpa pengetahuan pemilik peranti. Varian ini menggunakan aplikasi atau program yang tersembunyi di dalam peranti Android untuk mengambil alih kawalan peranti dan mengumpul data peribadi pengguna atau melancarkan serangan siber.

Varian ini dapat menyebar dengan cepat dan meluas di dalam jaringan komputer dan menjadi ancaman yang serius terhadap keselamatan siber dan privasi pengguna peranti Android. Teknik eksploitasi yang digunakan oleh varian ini bervariasi dan semakin berkembang seiring dengan perkembangan teknologi. Namun, tujuan utama varian *botnet* adalah untuk mengambil alih peranti Android yang terinfeksi, mencuri data peribadi, dan melakukan aktiviti jahat.

Penjelasan untuk beberapa varian mengikut tahun kemunculan adalah seperti berikut:

1. **Droiddream**: *Botnet* ini mula ditemui pada tahun 2011 dan menggunakan pemalsuan sijil digital dan kelemahan debug pada Android untuk menyebar dan menginfeksi peranti. Ia berupaya mengumpul data peribadi pengguna dan melakukan serangan DDoS.

2. **GinMaster**: GinMaster muncul pada tahun 2011 dan menggunakan teknik Man-in-the-Middle pada sambungan HTTPS untuk penyebaran ke peranti Android. *Botnet* ini berupaya untuk mencuri data peribadi dan memberikan akses jauh ke dalam peranti yang terinfeksi.

3. ***Geinimi***: *Botnet* ini ditemui pada tahun 2011 dan memasang dirinya pada aplikasi di luar Google Play. *Geinimi* boleh merakam panggilan telefon, mesej teks, dan mengumpul data peribadi pengguna.

4. **NotCompatible**: NotCompatible adalah jenis Android *Botnet* Varian yang ditemui pada tahun 2012. *Malware* ini tersebar melalui laman web yang memerangkap atau e-mel spam dan mencari kelemahan keselamatan pada peranti Android. NotCompatible boleh mengambil alih peranti Android dan menggunakannya sebagai server proxy untuk menghantar e-mel spam dan serangan rangkaian. *Malware* ini juga boleh memasang aplikasi tambahan pada peranti yang terinfeksi.

5. **Plankton**: Plankton muncul pada tahun 2012 dan menggunakan kombinasi nombor siri peranti dan PIN keyboard untuk menyebar dan menginfeksi peranti Android. *Botnet* ini boleh memberikan akses jauh ke dalam peranti dan mengumpul data peribadi.

6. **Obad**: *Botnet* ini muncul pada tahun 2013 dan menyamar sebagai aplikasi antivirus palsu untuk memasang dirinya pada peranti Android. Obad boleh

mencuri data peribadi, memberikan akses jauh ke dalam peranti, dan melakukan serangan DDoS.

7. **Krysanec:** Krysanec ditemui pada tahun 2014 dan menyebar melalui mesej teks, gambar, dan laman web. *Botnet* ini boleh mengumpul data peribadi dan memberikan akses jauh ke dalam peranti yang terinfeksi.

8. **Gooligan**: Gooligan adalah jenis Android *Botnet* Varian yang ditemui pada tahun 2016. *Malware* ini tersebar melalui aplikasi yang dimuat turun dari toko aplikasi pihak ketiga yang tidak dipercayai. Gooligan boleh mengambil alih peranti Android dan mencuri maklumat peribadi pengguna seperti e-mel dan kata laluan. *Malware* ini juga boleh memberikan akses yang tidak sah ke akaun Google pengguna, seperti Google Drive, Gmail, dan *Google Play.*

9. **BankBot**: BankBot adalah jenis Android *Botnet* Varian yang ditemui pada tahun 2017. *Malware* ini menyamar sebagai aplikasi perbankan dan tersebar melalui toko aplikasi pihak ketiga yang tidak dipercayai. BankBot boleh mengambil alih peranti Android dan mencuri maklumat perbankan pengguna seperti nombor kad kredit dan kata laluan. *Malware* ini juga boleh meniru antara muka aplikasi perbankan sebenar untuk memancing mangsa memberikan maklumat peribadi mereka.

10. `**WireX**: WireX muncul pada tahun 2017 dan mengambil alih peranti untuk melakukan serangan DDoS. *Botnet* ini mampu menginfeksi berbagai jenis peranti, termasuk peranti Android, dan menggunakan sumber daya peranti untuk melancarkan serangan.

11. **GhostCtrl**: *Botnet* ini muncul pada tahun 2017 dan menggunakan pemalsuan aplikasi popular seperti WhatsApp dan Pokemon Go untuk memasang dirinya pada peranti Android. GhostCtrl boleh memberikan akses jauh ke dalam peranti, mencuri data peribadi, dan melakukan serangan DDoS.

12.`**Judy**: Judy ditemui pada tahun 2017 dan menyebar melalui aplikasi permainan di Google Play. *Botnet* ini boleh memberikan akses jauh ke dalam peranti, mencuri data peribadi, dan melancarkan serangan DDoS.

13. **Rootnik**: Rootnik ialah varian Android *Botnet* yang ditemui pada 2016. *Malware* ini disebarkan melalui aplikasi palsu yang disebarkan melalui e-mel pancingan data. Rootnik boleh mengambil alih peranti Android dan menjadikannya sebahagian daripada rangkaian *botnet*. *Botnet* ini kemudiannya boleh digunakan untuk melancarkan serangan DDoS dan mencuri maklumat sensitif.

14. **Triada**: Triada ialah sejenis Varian Android *Botnet* yang ditemui pada 2016. Perisian hasad ini disebarkan melalui apl yang dimuat turun daripada gedung aplikasi pihak ketiga yang tidak dipercayai. Triada boleh mengambil alih peranti Android dan menukar sistem pengendalian untuk menyediakan akses yang lebih luas kepada peranti.

15. **Hummingbad**: Hummingbad ialah sejenis Varian Android *Botnet* yang ditemui pada tahun 2016. Perisian hasad ini merebak melalui apl yang dimuat turun daripada gedung apl pihak ketiga yang tidak dipercayai. Hummingbad boleh mengambil alih peranti Android dan mencipta rangkaian *botnet* yang boleh digunakan untuk memaparkan iklan palsu dan menjana pendapatan untuk penyerang.

16. **GhostCtrl**: GhostCtrl ialah sejenis Varian Android *Botnet* yang ditemui pada 2017. Perisian hasad ini merebak melalui apl yang dimuat turun daripada gedung apl pihak ketiga yang tidak dipercayai. GhostCtrl boleh mengambil alih peranti Android dan mengambil gambar dari kamera peranti, merakam suara dan mencuri data penting seperti maklumat kad kredit dan kata laluan.

17. **Xavier:** *Botnet* ini muncul pada tahun 2017 dan menyebar melalui iklan dalam aplikasi. Xavier boleh mengumpul data peribadi, memberikan akses jauh ke dalam peranti, dan melancarkan serangan DDoS.

18. **WireX**: WireX ialah varian Android *Botnet* yang ditemui pada 2017. Perisian hasad ini disebarkan melalui aplikasi palsu yang disebarkan melalui gedung aplikasi pihak ketiga. WireX boleh mengambil alih peranti Android dan menjadikannya sebahagian daripada rangkaian *botnet*. *Botnet* ini kemudiannya boleh digunakan untuk melancarkan serangan DDoS.

19. **Anubis**: Anubis ialah varian Android *Botnet* yang ditemui pada 2018. Perisian hasad ini disebarkan melalui apl palsu yang disebarkan melalui e-mel pancingan data dan tapak web palsu. Anubis boleh mengambil alih peranti Android dan mencuri maklumat sensitif seperti maklumat kad kredit dan kata laluan.

20. **Geost**: Geost ialah varian Android *Botnet* yang ditemui pada 2019. Perisian hasad ini disebarkan melalui apl yang dimuat turun daripada gedung apl pihak ketiga yang tidak dipercayai. Geost boleh mengambil alih peranti Android dan menjadikannya sebahagian daripada rangkaian *botnet*. *Botnet* ini kemudiannya boleh digunakan untuk melancarkan serangan DDoS dan mencuri maklumat sensitif.

21. **Gustuff:** Gustuff ialah sejenis Varian Android *Botnet* yang ditemui pada 2019. Perisian hasad ini disebarkan melalui aplikasi perbankan palsu yang disebarkan melalui halaman web palsu dan mesej teks pancingan data. Gustuff boleh mengambil alih peranti Android dan mencuri maklumat sensitif seperti maklumat log masuk dan kata laluan bank.

22. **Joker:** Joker ialah sejenis Varian Android *Botnet* yang ditemui pada 2019. Perisian hasad ini disebarkan melalui apl yang dimuat turun daripada gedung aplikasi pihak ketiga yang tidak dipercayai. Joker boleh mengambil alih peranti Android dan mencuri data penting seperti maklumat kad kredit dan kata laluan, serta menghantar mesej teks premium yang mahal tanpa kebenaran pengguna.

23. **Anserverbot**: *Anserverbot* pertama kali ditemukan pada tahun 2019. Dengan menggunakan *botnet* ini, penggodam dapat melancarkan serangan seperti serangan DDoS, mencuri maklumat sensitif, dan menyebarkan perisian berbahaya kepada peranti Android yang terinfeksi.

Jadual 2.1 menunjukkan senarai Android varian dan teknik eksploitasi dari 2011, tahun kemunculan *botnet* sehingga tahun 2022 yang diambil dari beberapa laporan daripada Miners (2014), Kumar (2016), Goodin (2017), Ramilli (2018), dan Korolov (2019).

Jadual 2.1 Senarai Varian *Botnet* untuk Android dari 2011 hingga 2022[3]

No	Varian	Tahun	Teknik Eksploitasi
1	Droiddream	2011	Melalui aplikasi yang dimuat turun dari luar Google Play Store
2	GinMaster	2011	Serangan Man-in-the-Middle pada jaringan HTTPS
3	Geinimi	2011	Perisian jahat diselitkan dalam aplikasi di luar Google Play
4	Plankton	2012	Kemasukan berdasarkan nombor siri peranti dan pin keyboard
5	Opfake	2012	Menyerupai aplikasi terkenal untuk menipu pengguna
6	BadNews	2013	Menggunakan nama aplikasi palsu
7	Obad	2013	Menyamar sebagai aplikasi antivirus palsu
8	Krysanec	2014	Penyebaran melalui mesej teks, gambar, dan laman web
9	Kemoge	2015	Menyebar melalui aplikasi yang dimuat turun dari luar Google Play dan spam
10	HummingBad	2016	Menyebar melalui aplikasi yang dimuat turun dari luar Google Play mesej spam
11	Rootnik	2016	Mengambil alih peranti dan menjadi sebahagian daripada rangkaian botnet
12	Anubis	2017	Melalui aplikasi palsu di luar Google Play Store
13	WireX	2017	Mengambil alih peranti untuk melakukan serangan DDoS
14	GhostCtrl	2017	Pemalsuan aplikasi popular seperti WhatsApp dan Pokemon Go
15	Judy	2017	Menyebarkan diri melalui aplikasi permainan di Google Play

bersambung…..

[3] Varian yang digariskan (underlined) menggunakan *URL* yang dienkripsi untuk penyebaran serangan

…..sambungan

16	Xavier	2017	Menyebarkan diri melalui iklan dalam aplikasi
17	Triada	2018	Menyebar melalui aplikasi yang dimuat turun dari luar Google Play Store
18	Gustuff	2018	Menyebar melalui iklan dalam aplikasi dan menggunakan
19	Ginp	2019	Menyebar melalui aplikasi perbankan
20	Joker	2019	Menyebar melalui aplikasi yang dimuat turun dari Google Play Store dan spam
21	BlackRock	2020	Menyebar melalui aplikasi phishing dan mampu mencuri kredensial pengguna
22	FluBot	2021	Menyebar melalui mesej Inggeris dan mampu mencuri kredensial perbankan
23	Alien	2022	Menyebar melalui aplikasi yang dimuat turun dari luar Google Play Store dengan nama palsu

Dari 23 varian yang dilampirkan di **Jadual 2.1**, terdapat 6 varian yang menggunakan *URL* yang dienkripsi untuk penyebaran serangan: *Droiddream*, *Geinimi*, *Opfake*, *Kemoge*, *HummingBad*, *Triada*, dan *FluBot*. Berikut adalah beberapa masalah yang dihadapi dalam mengesan keenam-enam varian ini:

1. Enkripsi *URL*: Varian-varian ini menggunakan enkripsi pada *URL* untuk menyembunyikan atau melindungi komunikasi antara peranti terinfeksi dan pelayan pengawal. Enkripsi ini membuat *URL* sukar untuk dianalisis dan menghalang pengesanan.

2. Pola Komunikasi yang Berubah-ubah: Varian-varian ini sering kali mengubah pola komunikasi mereka untuk mengelakkan pengesanan. Contohnya, varian ini boleh menggunakan rangkaian C&C yang berbeza, atau menggunakan teknik yang baru untuk mengubah cara varian ini berkomunikasi secara dinamik.

3. Pola Tindakan yang Sama dengan Aplikasi Sah: Beberapa varian *botnet* ini dapat menyamar sebagai aplikasi sah atau menggunakan pola tindakan yang sama dengan aplikasi sah untuk mengelakkan pengesanan. Ini menyukarkan

penganalisa untuk membezakan antara perilaku *botnet* dan aplikasi yang sebenarnya.

4. Keupayaan Mengelakkan Pengesanan: Varian-varian ini sering kali mempunyai mekanisme untuk mengelakkan pengesanan, seperti mengelakkan pengesan antivirus dan mengelirukan alat pengesanan.

5. Evolusi dan Peningkatan: *Botnet* ini terus berkembang dan meningkatkan keupayaan mereka dari masa ke masa. Mereka boleh mengubah teknik dan strategi mereka untuk mengelakkan pengesanan dan melindungi diri daripada tindakan pencegahan.

Kesemua masalah ini membuatkan pengesanan dan pencegahan terhadap keenam-enam varian ini menjadi cabaran yang kompleks dalam usaha melawan ancaman *botnet*. Fokus utama untuk projek ini adalah untuk menyelesaikan masalah pertama iaitu enkripsi *URL*.

2.3 TINJAUAN LITERATUR (AKADEMIK DAN INDUSTRI)

Pengesanan *botnet* bagi peranti Android yang melibatkan penggunaan enkripsi sebagai teknik serangan telah menjadi fokus utama para pengkaji dari bidang akademik mahupun industri sejak kemunculan *botnet* Android yang pertama pada March 2011 dengan varian yang dikenali sebagai *Droiddream* (Bilge et al. 2011). Ini disusuli dengan beberapa kajian seperti analisis *botnet URL* (Abdul Kadir et al. 2015), analisis aliran tingkah laku *botnet* (Xu et al, 2014), dan penggunaan *machine learning* (Wang et al., 2020, Rai et al. 2022). Terdapat beberapa kajian literatur termasuk dari bidang akademik dan industri yang telah dirujuk untuk membuat kajian ini. Rumusan berkaitan kelebihan dan kelemahan berkaitan kajian-kajian terdahulu adalah seperti yang dilampirkan di **Jadual 2.2**, **Jadual 2.3,** dan **Jadual 2.4.**

Botnet merupakan ancaman besar bagi keamanan jaringan, dengan *server command and control (C&C)* menjadi target utama pengesanan. Salah satu pendekatan untuk mengesan server C&C tersebut adalah melalui analisis *netflow (sequence of*

packet), yang digunakan untuk memantau dan menganalisis trafik jaringan seperti yang ditunjukkan dalam kajian Li et al. (2013), Zhao et al. (2013), dan Bilge et al. (2011), di mana teknik ini dapat mengesan ciri-ciri komunikasi *botnet* (C&C) dengan mengukur aliran jaringan. Namun, kajian ini cuma dapat digunakan dalam persekitaran yang mempunyai aliran jaringan dan sukar untuk mengesan jenis serangan *botnet* yang lebih kompleks.

Suárez-Tangil et al. (2013) pula memberikan gambaran tentang evolusi dan pengesanan *malware* yang lebih fokus kepada AI, termasuklah *Android Botnet*. Wang et al. (2013) memberikan gambaran menyeluruh tentang karakteristik, vektor serangan, dan teknik pengesanan *botnet* Android. Pada tahun 2014, Xu et al. pula mencadangkan teknik analisis tingkah laku aliran untuk mengesan trafik *botnet* yang disulitkan. Kajian-kajian ini telah menyediakan gambaran umum mengenai evolusi dan pengesanan perisian hasad yang memfokuskan kepada peranti pintar. Walau bagaimanapun, terdapat beberapa kelemahan seperti: i) tidak melibatkan kajian empirikal dan tidak menawarkan teknik baru untuk mengesan *botnet*. ii) tidak memberikan teknik pengesanan baru untuk Android *Botnet URL*s, hanya menyediakan tinjauan terhadap kaedah yang sedia ada. iii) teknik ini memerlukan data latihan yang banyak, hasil analisis kurang tepat ketika menangani dataset yang tidak diketahui sebelumnya.

Abdul Kadir et al. (2015) dalam kajian mereka mendapati bahawa *botnet* menggunakan pelbagai teknik untuk mengaburkan komunikasi mereka, seperti menggunakan alamat IP dan bukannya nama domain, dan menggunakan port bukan standard. Penulis juga mengenal pasti beberapa ciri *URL botnet* yang boleh digunakan untuk pengesanan, seperti panjang *URL*, kehadiran kata kunci tertentu dan penggunaan aksara luar biasa. Walau bagaimanapun, kajian ini tidak melampirkan kerangka analisis tapi hanya membincagkan tingkah laku *URL botnet*. Jumlah data yang diambil dalam kajian ini terbatas, tidak merangkumi semua jenis *botnet* Android di luar sana. Penyelidikan tambahan diperlukan untuk membangunkan teknik pengesanan yang lebih berkesan dan mantap yang boleh digeneralisasikan kepada rangkaian *botnet* Android yang lebih luas.

Kajian ini kemudian diikuti oleh Chung et al. (2016) yang memperkenalkan pendekatan pengesanan *botnet* Android berdasarkan analisis tingkah laku rangkaian, menggunakan algoritma *machine learning* untuk menganalisis trafik rangkaian. Tetapi kajian ini hanya menganalisis tingkah laku rangkaian dan tidak menyertakan analisis fail APK. Dataset yang digunakan juga tidak mencerminkan kepelbagaian jenis serangan *botnet*.

Tso et al. (2018) menyediakan semakan komprehensif teknik pengesanan *botnet* untuk platform Android, menyerlahkan cabaran dan jurang penyelidikan dalam bidang tersebut. *Botnet* semakin menggunakan penyulitan untuk mengaburkan saluran komunikasi mereka. Namun begitu, Tso et al. (2018) tidak melibatkan kajian empirikal dan tidak menghasilkan teknik baru untuk mengesan *botnet*.

Wang, Yu, dan Zhu (2020) juga tidak menggunakan dataset khusus dalam penelitian mereka tentang pengesanan *botnet* menggunakan *machine learning*. Sebaliknya, mereka menggunakan dataset dari kajian sebelumnya yang berkaitan dengan pengesanan *botnet*. Pada tahun yang sama juga, kajian Alshammari dan Khan (2020) memberikan kajian sistematik mengenai algoritma *machine learning* yang digunakan dalam pengesanan *botnet*. Kajian ini memberi gambaran yang jelas mengenai kelebihan dan kelemahan setiap teknik dan dapat membantu dalam memilih teknik yang paling sesuai untuk keperluan pengesanan *botnet*. Pada tahun yang sama juga, Wang, Yu, dan Zhu (2020) mengkaji penggunaan *machine learning* untuk pengesanan *botnet* secara umum. Tetapi, kajian-kajian ini adalah terhad pada teknik dan penggunaan *machine learning.*

Terkini, Rai, Yadav, Singh telah (2022) mengembangkan pendekatan hibrid untuk mengesan *botnet* Android menggunakan gabungan teknik *machine learning*. Kajian ini memberi tumpuan khusus kepada pengesanan *botnet* pada platform Android. Walau pun begitu, kajian ini tidak dapat mengesan *botnet* dengan ciri-ciri unik dan baru yang tidak dikenal pasti sebelumnya.

Di samping itu, terdapat beberapa kajian yang berkaitan *botnet* yang menggunakan teknik visualisasi. Kajian pertama pada tahun 2014 oleh J. Garcia et al.

adalah dengan menggunakan analisis jaringan sosial sebagai teknik visualisasi untuk menggambarkan struktur dan perilaku *botnet*. Kajian ini memberikan gambaran tentang bagaimana *botnet* berinteraksi dalam jaringan.

Ini disusuli oleh kajian S. Patil dan N. Khochare (2015) yang menggunakan sistem informasi geografis (GIS) sebagai teknik visualisasi untuk memetakan penyebaran *botnet* secara geografi. Dengan memanfaatkan GIS, kajian ini dapat memberikan gambaran tentang persebaran *botnet* mengikut kedudukan geografi. Tetapi, fokus kajian ini terbatas pada visualisasi geografi sahaja dan tidak membahas aspek teknik analisis *botnet* secara mendalam.

Seterusnya, pada tahun 2016 kajian M. H. Ahmed et al. menggunakan analisis aliran data sebagai teknik visualisasi untuk menggambarkan *botnet intrusion*. Kajian ini bertujuan untuk mengesan dan memahami serangan *botnet* melalui analisis data aliran jaringan.

Pada tahun 2017, kajian oleh M. M. R. Islam et al. menggunakan analisis lalu lintas jaringan sebagai teknik visualisasi untuk menggambarkan pola komunikasi dan hubungan dalam *botnet*. Melalui analisis lalu lintas jaringan, kajian ini mampu mengesan pola komunikasi yang mencurigakan dan hubungan antara entiti di dalam *botnet*. Namun, kajian ini tidak memberikan penjelasan mendalam tentang teknik visualisasi yang digunakan.

H. Chen et al. (2018) menggabungkan teknik visualisasi dan analitik untuk mengesan dan menyelidiki *botnet*. Kajian ini fokus pada pengembangan solusi visual yang efektif untuk menganalisis aktiviti *botnet*.

Seterusnya, kajian oleh N. Firdaus et al. (2018) menggunakan analisis lalu lintas DNS sebagai teknik visualisasi untuk mengesan dan menggambarkan aktiviti *botnet* melalui lalu lintas DNS. Melalui teknik ini, kajian ini dapat menjelaskan pola dan hubungan antara domain yang terlibat dalam operasi *botnet*. Namun, kajian ini tidak memberikan penjelasan terperinci tentang teknik visualisasi yang digunakan.

Kajian oleh A. Yadav et al. (2019) membentangkan teknik visualisasi untuk menganalisis ancaman *botnet* dalam jaringan skala besar. Meskipun tidak dijelaskan secara terperinci tentang teknik visualisasi yang digunakan, kajian ini memberikan sumbangan dalam memahami dan mengesani ancaman *botnet* pada skala yang luas.

Kajian oleh S. Shin et al. (2019) menggunakan metrik graf sebagai teknik visualisasi untuk menganalisis aktiviti *botnet*. Kajian ini memanfaatkan analisis graf untuk menggambarkan hubungan dan pola komunikasi antara entiti dalam *botnet*.

Zhang et al. (2020) menggunakan visualisasi khusus *Internet of Things (IoT)* untuk menggambarkan serangan *botnet* pada perangkat IoT. Kajian ini bertujuan untuk meningkatkan pemahaman tentang serangan *botnet* pada lingkungan IoT.

Pada tahun yang sama di 2020, kajian oleh K. R. Ray et al. memanfaatkan analisis *time-series* dan teknik visualisasi untuk menggambarkan pola komunikasi *botnet*. Kajian ini dapat mengesan perilaku dan pola komunikasi *botnet* secara temporal. Namun, kajian ini tidak membahas secara terperinci tentang teknik analisis yang digunakan.

Jadual 2.2 Kajian literatur dari bidang akademik[4]

No.	Penulis	Teknik Kajian	Dataset	Kesimpulan
1	Bilge et al. (2011)	Analisa Netflow	Real-world dataset	Dapat mengesan ciri-ciri komunikasi *botnet* (C&C) dengan mengukur aliran jaringan
2	Suárez-Tangil et al. (2013)	Tinjauan terhadap perisian jahat	Kajian literatur	Menyediakan gambaran umum mengenai evolusi dan pengesanan perisian hasad yang memfokuskan kepada peranti pintar
3	Wang et al. (2013)	Tinjauan terhadap *botnet* Android	Kajian literatur	Memberikan gambaran menyeluruh tentang karakteristik, vektor serangan, dan teknik pengesanan *botnet* Android
4	Xu et al. (2014)	Analisis Aliran Tingkah Laku	Dataset sintetik	Teknik baru untuk mengesan lalu lintas *botnet* yang dienkripsi

bersambung…..

[4] Penulis yang digariskan (underlined) meggunakan teknik analisis URL dalam kajian

…..sambungan

5	Abdul Kadir et al. (2015)	Analisis *URL* pengumpul sample	Data sampel Android *botnet URL*s	Memeriksa tahap penerimaan *botnet* Android ke rangkaian web, mengenal pasti *URL-URL* yang paling sering digunakan oleh *botnet* dan melihat trend terbaru
6	Chung et al. (2016)	Analisis Tingkah Laku Jaringan	Dataset aplikasi	Menyediakan pendekatan pengesanan *botnet* yang dapat mengenal pasti jenis serangan dan mempunyai ketepatan yang tinggi
7	Tso et al. (2018)	Tinjauan komprehensif	Kajian literatur	Menganalisis teknik-teknik pengesanan *botnet* untuk platform Android dan menyoroti cabaran kajian dalam bidang ini
8	Wang, X., Yu, F., dan Zhu, Y. (2020)	Machine Learning	-	Kesan cepat dan ketepatan yang lebih baik dalam mengesan *botnet*
9	Alshammari, R. dan Khan, S. U. (2020)	Kajian tinjauan sistematis	-	Menyediakan gambaran keseluruhan teknik pembelajaran mesin dalam mengesan *botnet*
10	Foozy, C. F. M., Wen, C. C., & Chinniah, M. (2020)	Klasifikasi	Dataset milik sendiri	Model yang dihasilkan memiliki ketepatan yang tinggi.
11	Sari, A. A. dan Ibrahim, S. (2021)	Kajian literatur	-	Memberikan gambaran menyeluruh tentang teknik-teknik yang terdapat untuk mengesan *botnet*
12	Rai et al. (2022)	Hybrid dan Klasifikasi	Data perbankan tiruan	Keberkesanan dan kecekapan yang tinggi dalam mengesan *botnet*

Jadual 2.3 Kajian literatur dari bidang akademik berkaitan visualisasi

No.	Penulis (Tahun)	Tajuk Kajian	Teknik Visualisasi	Kesimpulan
1	J. Garcia et al. (2014)	*Visualizing Botnet Structures and Behaviors*	Social Network Analysis	Memvisualisasikan struktur dan perilaku *botnet* menggunakan analisis jaringan sosial.
2	S. Patil dan N. Khochare (2015)	*Visualizing the Spread of Botnets using GIS*	Sistem Informasi Geografis (GIS)	Memetakan penyebaran *botnet* secara geografis.
3	M. H. Ahmed et al. (2016)	*Visualizing Botnet Intrusions through Flow Data Analysis*	Flow-based Visualization	Menggunakan analisis aliran data sebagai teknik visualisasi untuk memvisualisasikan intrusi *botnet*.

bersambung…

…..sambungan

4	M. M. R. Islam et al. (2017)	*Visualizing Botnet Behavior using Network Traffic Analysis*	Analisis lalu lintas jaringan	Menggambarkan pola komunikasi dan hubungan dalam *botnet*
5	H. Chen et al. (2018)	*Visual Analytics for Botnet Detection and Investigation*	Visual Analytics	Menggabungkan teknik visualisasi dan analitik untuk mendeteksi dan menyelidiki *botnet*.
6	N. Firdaus et al. (2018)	*Visualizing Botnet Activities through DNS Traffic Analysis*	Analisis lalu lintas DNS	Mengidentifikasi aktivitas *botnet* melalui lalu lintas DNS. DNS
7	A. Yadav et al. (2019)	*Visualizing Botnet Threats in Large-Scale Networks*	Tidak dijelaskan	Menganalisis ancaman *botnet* dalam jaringan skala besar.
8	S. Shin et al. (2019)	*Visual Analysis of Botnet Activities using Graph Metrics*	*Graph-based Visualization*	Memanfaatkan analisis graf dan metrik graf untuk memvisualisasikan aktivitas *botnet*.
9	L. Zhang et al. (2020)	*Visualizing Botnet Attacks on IoT Devices*	*Internet of Things (IoT) Visualization*	Menggunakan visualisasi khusus IoT untuk memvisualisasikan serangan *botnet* pada perangkat IoT.
10	K. R. Ray et al. (2020)	*Visualizing Botnet Communication Patterns using Time-Series*	Analisis *Time-Series* dan teknik visualisasi	Memvisualisasikan pola komunikasi *botnet* melalui analisis deret waktu.

Jadual 2.4 Kajian literatur dari bidang industri

No	Penyelesaian	Fitur	Pengesanan *Botnet URL*	Pengesanan *URL* yang Dienkripsi
1	Symantec Mobile (2023)	Pengesanan dan perlindungan ancaman canggih dan menggunakan algoritma pembelajaran mesin dan analisis tingkah laku	Ya	Ya
2	McAfee Mobile Security (2023)	Menggunakan pembelajaran mesin dan teknologi AI dan menyediakan pengesanan dan perlindungan *botnet*	Ya	Ya
3	Lookout Mobile Security (2023)	Menggunakan analisis tingkah laku untuk mengenal pasti aktiviti *botnet* dan menyediakan pengesanan dan perlindungan *botnet*	Ya	**Tidak**
4	Kaspersky Internet Security untuk Android (2023)	Menyediakan perlindungan terhadap *botnet* dan ancaman lain dan menggunakan kombinasi analisis berdasarkan *signature*	Ya	**Tidak**

bersambung…..

…..sambungan

5	Bitdefender Mobile Security & Antivirus (2023)	Menyediakan pengesanan *botnet* dan ciri keselamatan lain	Ya	**Tidak**
6	Norton Mobile Security (2023)	Menyediakan perlindungan terhadap *botnet* dan ancaman lain dan menggunakan analisis tingkah laku untuk mengesan dan mencegah serangan *botnet*	Ya	**Tidak**

2.4 PERBANDINGAN SISTEM SEDIA ADA

Setiap kajian (akademik) dan juga sistem keselamatan mudah alih (industri) mempunyai kekuatan dan kelemahan masing-masing. Meskipun terdapat beberapa kajian akademik sebelum ini seperti Bilge et al. (2011), Suárez-Tangil et al. (2013), Wang et al. (2013), Xu et al. (2014), Abdul Kadir et al. (2015), Chung et al. (2016), Tso et al. (2018), Wang, X., Yu, F., dan Zhu, Y. (2020), Alshammari, R. dan Khan, S. U. (2020), Foozy, C. F. M., Wen, C. C., & Chinniah, M. (2020), Sari, A. A. dan Ibrahim, S. (2021), dan Rai et al. (2022)., masih terdapat jurang kajian yang boleh dilengkapkan, seperti berikut:

1. **Kurangnya Pemahaman Mendalam**: Walaupun kajian-kajian tersebut memberikan pandangan awal tentang penggunaan teknik enkripsi dan visualisasi dalam menganalisis pautan rangkaian *botnet* Android, kebanyakan daripadanya masih terhad kepada penerokaan permulaan. Terdapat keperluan untuk kajian yang lebih mendalam dan terperinci untuk memahami secara menyeluruh bagaimana teknik enkripsi digunakan dalam *botnet* Android, bagaimana visualisasi dapat membantu dalam menganalisis pautan, dan juga untuk menyelidik kesan teknik ini terhadap keberkesanan pengesanan dan pencegahan.

2. **Keperluan Penyelidikan Lanjutan**: Walaupun terdapat beberapa kajian yang terkini, seperti Sari dan Ibrahim (2021) dan Rai et al. (2022), masih terdapat keperluan untuk penyelidikan lanjutan yang melibatkan lebih banyak variasi varian *botnet* Android. Ini akan membantu dalam mengukur keberkesanan teknik enkripsi dan visualisasi dalam mengesan dan menganalisis jaringan *botnet* yang lebih pelbagai dan kompleks.

3. **Integrasi dengan Sistem Keselamatan Mudah Alih**: Kajian-kajian ini umumnya memberi tumpuan kepada aspek akademik dan kurang memberikan penekanan yang mencukupi kepada aspek implementasi praktikal dalam sistem keselamatan mudah alih.

Terdapat keperluan untuk mengkaji secara holistik dan menyeluruh bagaimana teknik enkripsi dan visualisasi dapat diintegrasikan ke dalam sistem keselamatan mudah alih yang ada dan memberikan impak yang signifikan dalam memerangi ancaman *botnet*.

4. **Keselamatan dan Privasi**: Kajian-kajian ini juga perlu memberi perhatian yang lebih kepada kesan penggunaan teknik enkripsi dan visualisasi terhadap keselamatan dan privasi pengguna. Terdapat keperluan untuk mengkaji isu-isu yang berkaitan dengan kerahsiaan data yang dihurai dan bagaimana melindungi privasi individu dalam konteks analisis *botnet*.

Dari perspektif industri pula, terdapat beberapa contoh sistem keselamatan mudah alih yang sedia ada di pasaran seperti Symantec Mobile, McAfee Mobile Security, Lookout Mobile Security, Kaspersky Internet Security untuk Android, Bitdefender Mobile Security & Antivirus, dan Norton Mobile Security. Berdasarkan **Jadual 2.4**, terdapat beberapa jurang kajian yang telah dikenal pasti:

1. Walaupun kebanyakan penyelesaian yang disenaraikan menyediakan pengesanan *botnet URL*, hanya Symantec dan McAfee yang dinyatakan mempunyai keupayaan pengesanan *URL* yang dienkripsi (*VirusTotal*, 2023).

2. Walaupun Lookout Mobile Security menggunakan analisis tingkah laku untuk mengesan aktiviti *botnet*, beberapa pengguna melaporkan masalah positif palsu.

3. Kebanyakan penyelesaian yang disenaraikan tidak menyebut secara khusus mengenai keupayaan pengesanan *URL* terenkripsi.

4. Kajian yang dijalankan oleh Abdul Kadir et al. (2015) mendapati ada beberapa sistem keselamatan mudah alih dan juga *blacklist* yang tidak dapat mengesan teknik enkripsi yang telah diubah suai mengikut varian tertentu. Hal ini dibuktikan melalui dapatan kajian dimana tiada padanan *URL*s yang merbahaya di dalam senari *blacklist malware URLs*.

2.5 KESIMPULAN

Terdapat keperluan untuk penyelidikan dan pembangunan penyelesaian pengesanan *botnet* yang dapat mengesan *URL* yang dienkripsi dengan lebih cekap; dan mengoptimumkan analisis tingkah laku untuk mengurangkan jumlah positif palsu dan meningkatkan keberkesanan pengesanan *botnet*. Untuk mengisi jurang-jurang kajian yang dinyatakan, projek ini membentangkan satu rangka kerja analisis untuk *URL botnet* Android dengan menggunakan teknik enkripsi dan visualisasi.

Projek ini memfokuskan kepada kajian yang lebih mendalam untuk memahami *botnet* varian secara menyeluruh. Contohnya, bagaimana teknik enkripsi digunakan dalam *botnet* Android, bagaimana visualisasi dapat membantu dalam menganalisis pautan, dan juga untuk menyelidik kesan teknik ini terhadap keberkesanan pengesanan dan pencegahan. Projek ini diharapkan dapat memberikan pemahaman baru, penyelesaian yang berkesan, dan sumbangan yang signifikan dalam domain analisis pautan rangkaian Android Bot.

BAB III

KAEDAH KAJIAN

3.1 PENGENALAN

Bab ini menjelaskan kaedah yang diguna dalam menjalankan kajian. Sub topik bab ini dibahagikan kepada dua: 1) metodologi kajian yang akan menerangkan empat fasa iaitu analisis, reka bentuk, pembangunan, dan penilaian. 2) reka bentuk kerangka yang menjelaskan secara terperinci berkaitan rangka kerja analisis untuk pautan rangkaian android bot (android *botnet url*s) dengan menggunakan teknik enkripsi dan visualisasi.

3.2 METODOLOGI KAJIAN

Kajian ini dilakukan dalam empat fasa seperti yang dilampirkan dalam **Rajah 3.1**:

1. Dalam fasa pertama (**Analisis**), pengkaji melakukan semakan literatur yang komprehensif mengenai kajian sedia ada mengenai *botnet* dan keselamatan Android. Ini akan membantu untuk mengenal pasti teknik dan alat terkini yang digunakan oleh pengendali *botnet* dan keadaan terkini dalam pengesanan *botnet* Android.

2. Dalam fasa kedua (**Rekabentuk**), pengkaji mengumpul satu set data besar peranti Android termasuk *URL*s. Kemudian, pengkaji akan menggunakan

algoritma untuk menganalisis *URL*s dan mengenal pasti corak yang mungkin menunjukkan penggunaan enkripsi di dalam *botnet* Android.

3. Di dalam fasa ketiga (**Pembangunan**), pengkaji merekabentuk dan melaksanakan prototaip rangka kerja pengesanan. Rangka kerja ini menggabungkan algoritma yang dibangunkan dalam fasa kedua dan dapat mengenal pasti *URL botnet* Android yang dienkripsi.

4. Dalam fasa keempat (**Penilaian**), pengkaji menilai prestasi rangka kerja pengesanan dan membandingkan rangka kerja yanga dibangunkan dengan kaedah pengesanan *botnet* Android sedia ada untuk menentukan keberkesanannya.

Rajah 3.1 Empat fasa kajian

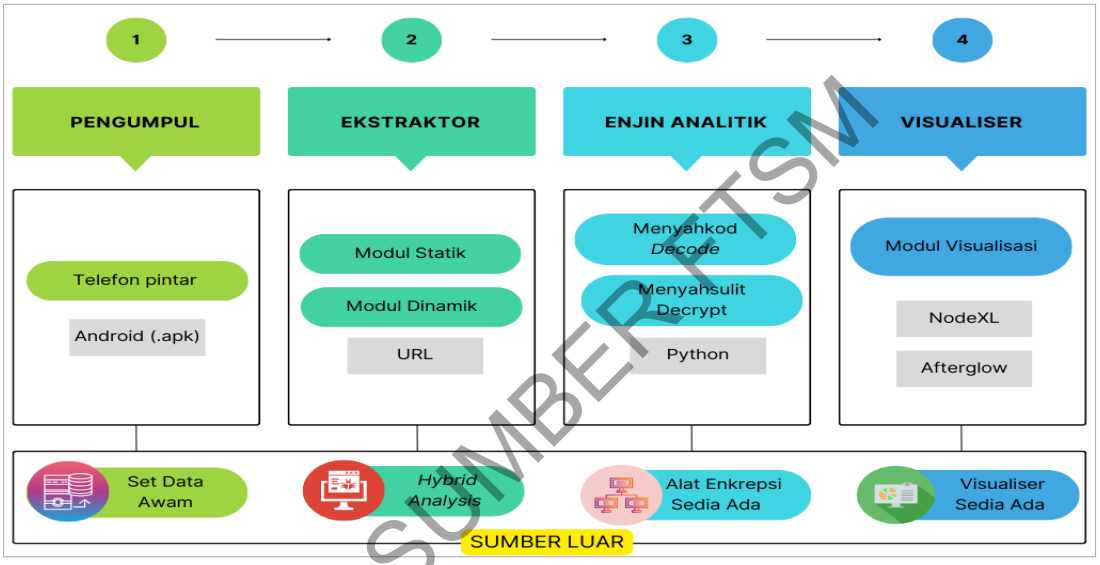
3.3 REKA BENTUK KERANGKA

Kajian ini memperkenalkan satu kerangka analisis yang dihasilkan sendiri berpandukan beberapa kerangka siber keselamatan yang telah dibentangkan oleh *Canadian institute for Cybersecurity (CIC)*[5]. Rangka kerja ini bertujuan untuk mengatasi cabaran dalam

[5] https://www.unb.ca/cic/research/index.html

mengesan dan menganalisis dengan tepat pautan rangkaian yang dienkripsi yang digunakan oleh *botnet* Android.

Dengan menggabungkan teknik enkripsi dan visualisasi, kajian ini memberi penekanan kepada penyelesaian yang dapat mendekripsi dan menganalisis pautan tersebut, membantu dalam memahami dengan lebih baik aktiviti *botnet* dan meningkatkan keberkesanan pengesanan terhadap ancaman *botnet* di platform Android.



Rajah 3.2 Kerangka analisis pautaan rangkaian Android Bot (Android *Botnet URL*s)

Kerangka kerja ini terdiri daripada empat tahap: Pengumpul, Ekstraktor, Enjin Analitik, dan Visualiser seperti yang dilampirkan dalam **Rajah 3.2**. Penjelasan setiap fasa adalah seperti yang berikut:

1. **Pengumpul**: Fasa pertama dalam kerangka kerja pengesanan Android *botnet URL* adalah pengumpulan data. Di sini, data seperti *botnet apk* dan *URLs* yang berkaitan dengan Android *botnet* dikumpul dari set data awam seperti dari blog dan laman sesawang *contagio dump*[6], *Canadian Institute for Cybersecurity*

[6] http://contagiominidump.blogspot.com/

(CIC)[7], dan *VirusTotal*[8]. Data ini harus mempunyai pelbagai varian. Pengumpulan data yang memadai sangat penting untuk membolehkan analisis untuk menyahkod dan menyahsulit adalah tepat.

2. **Ekstraktor**: Fasa kedua adalah ekstraksi data. Di dalam fasa ini, data yang telah dikumpulkan dari fasa satu akan diolah untuk mengenal pasti corak tingkahlaku yang menunjukkan kehadiran enkripsi di dalam *URL botnet* Android seperti penggunaan algoritma yang sudah diubahsuai mengikut varian tertentu. Corak ini kemudian akan diekstrak untuk analisis seterusnya.

3. **Enjin Analitik**: Fasa ketiga adalah fasa analisis. Di sini, data yang diekstrak akan dianalisis menggunakan algoritma menyahkod dan menyahsulit untuk mengenal pasti *URL botnet* Android yang dienkripsi. Hasil analisis ini kemudian akan dihantar ke fasa seterusnya.

4. **Visualiser**: Fasa keempat adalah visualisasi. Di sini, hasil analisis akan disajikan dengan cara yang mudah difahami oleh pengguna. Sebagai contoh, *URL botnet* Android yang dikesan akan dipaparkan dalam senarai yang mudah dibaca dan dilihat dengan menggunakan alat visualisasi seperti *NodeXL*[9] dan juga *Afterglow*[10]. Visualisasi data adalah penting untuk membolehkan pengguna menilai dengan mudah kesan kerja kerangka kerja pengesanan *botnet*. Kedua-dua alat visualisasi ini mudah digunakan berbanding *Gephi*, *Tableau*, dan *Cytoscape*. *NodeXL* menyediakan fungsi analisis jaringan yang berguna untuk mengenalpasti struktur dan corak dalam data. Integrasi yang baik dengan *Excel*, membolehkan penggunaan data sedia ada dan analisis statistik. *Afterglow* pula membolehkan visualisasi dengan fungsi-fungsi yang fleksibel.

[7] https://www.unb.ca/cic/datasets/index.html

[8] https://www.virustotal.com/

[9] https://www.smrfoundation.org/nodexl/

[10] https://afterglow.sourceforge.net/

Kerangka analisis ini dihasilkan melalui langkah-langkah berikut:

1. **Pengenalanan Masalah**: Pada langkah awal, masalah utama yang dihadapi oleh solusi-solusi sedia ada, iaitu ketidakmampuan untuk mengesan dengan tepat *botnet* Android yang menggunakan enkripsi pada *URL*, dikenal pasti dan dianalisis secara terperinci.

2. **Kajian Literatur**: Tahap ini melibatkan penyelidikan literatur yang komprehensif untuk memahami kerangka analisis yang telah dibangunkan sebelum ini, teknik enkripsi yang digunakan dalam *botnet* Android, dan alat-alat visualisasi yang berkaitan.

3. **Reka Bentuk Kerangka Analisis**: Berdasarkan pemahaman daripada kajian literatur, sebuah kerangka analisis direka bentuk. Kerangka ini terdiri daripada empat fasa utama. Setiap fasa mempunyai fungsi khusus dalam proses analisis *URL botnet* Android yang terenkripsi.

4. **Implementasi dan Ujian**: Setelah perancangan, kerangka analisis ini diimplementasikan sebagai perisian menggunakan bahasa pengaturcaraan yang sesuai, seperti *Python*. Kemudian, kerangka analisis diuji menggunakan sampel *URL botnet* Android yang terenkripsi untuk memastikan keberkesanan dan ketepatan analisis. Dalam fasa ini, hanya tiga (3) varian *botnet* (*Anserverbot*, *PJapps*, dan *Droiddream*) yang digunakan untuk menguji kerangka analisis yang dicadangkan. Ketiga-tiga varian *botnet* yang dipilih merupakan ancaman yang signifikan atau berkaitan dengan serangan yang telah dilaporkan secara meluas. Menguji kerangka analisis pada varian *botnet* yang relevan dapat memberikan maklumat yang berharga tentang keberkesanan kerangka tersebut dalam menghadapi ancaman di dalam siber.

5. **Penilaian Keputusan**: Langkah ini melibatkan perbandingan keputusan analisis dengan keadaan sebenar dan pengukuran tahap kejayaan dalam mendekripsi dan menguraikan *URL botnet* Android yang terenkripsi. Penilaian ini dilakukan dengan membuat perbandingan *blacklist* dan *whitelist* dari

sumber luar seperti *Hybrid Analysis* [11] dan *VirusTotal*[12]. Dalam penilaian ini, data dari sumber luar digunakan untuk mengesahkan dan membandingkan *URL* yang dikenal pasti sebagai sebahagian daripada Android *Botnet*. Dengan menggunakan maklumat dari sumber-sumber tersebut, penilaian dijalankan untuk mengenal pasti sama ada *URL* tersebut termasuk dalam senarai hitam yang mencurigakan atau senarai putih yang selamat. Melalui perbandingan ini, dapat dinilai tahap kepercayaan dan potensi ancaman yang berkaitan dengan *URL* yang dienkripsi dan diubahsuai, serta membantu dalam proses analisis dan membuat keputusan lanjut yang berkaitan dengan *URL* Android *Botnet*.

3.4 KESIMPULAN

Penerangan mengenai varian *botnet* untuk Android telah memberikan pemahaman yang mendalam mengenai pelbagai jenis *botnet* yang terdapat dalam ekosistem Android. Terdapat pelbagai varian *botnet* yang direka khas untuk menyerang peranti Android, dan pemahaman ini adalah penting untuk mengenal pasti dan menghadapi ancaman yang mungkin dihadapi. Tinjauan literatur yang dilakukan, sama ada daripada sumber akademik atau industri, telah memberikan gambaran mengenai penyelidikan dan perkembangan terkini dalam domain *botnet* Android. Kajian ini menggunakan pengetahuan sedia ada untuk memahami dan menganalisis ancaman yang berkaitan dengan *URL botnet* Android yang dienkripsi dan diubahsuai.

Selain itu, bab ini juga melibatkan perbandingan sistem-sistem yang sedia ada untuk menangani *botnet* Android. Perbandingan ini membolehkan pengenalpastian kelebihan dan kelemahan sistem-sistem tersebut, memberi landasan untuk reka bentuk dan pembangunan kerangka analisis dan visualisasi yang akan dilakukan dalam kajian ini. Maklumat ini akan digunakan sebagai asas untuk meneruskan kepada bab-bab seterusnya dalam kajian ini.

[11] https://www.hybrid-analysis.com/

[12] https://www.virustotal.com/gui/

BAB IV

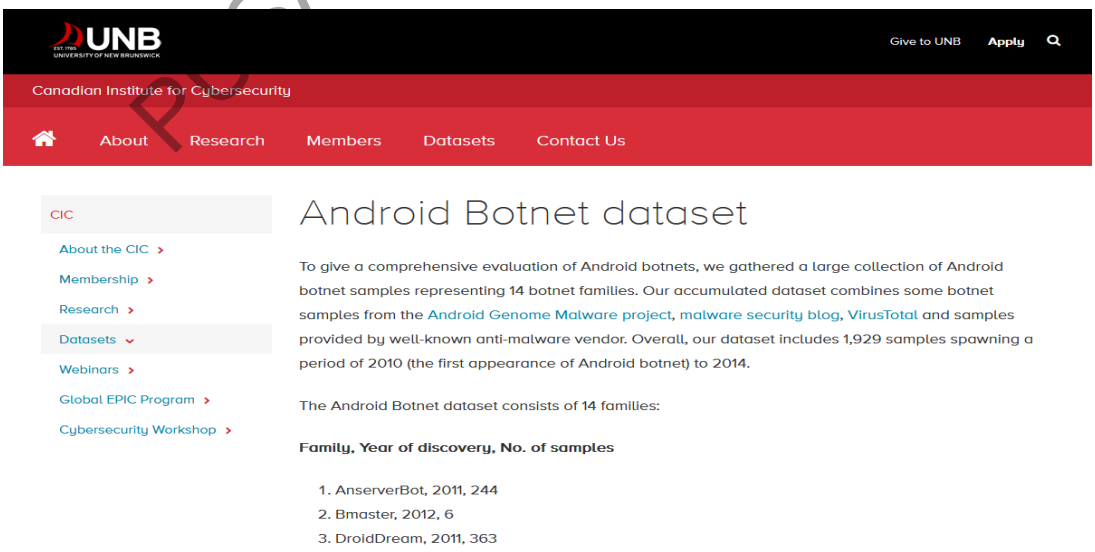
PELAKSANAAN KAJIAN

4.1 PENGENALAN

Bab ini membincangkan pelaksanaan kajian yang terdiri daripada beberapa sub-topik: pengumpulan data, pembangunan kerangka analisis, dan pengaturcaraan dan algoritma.

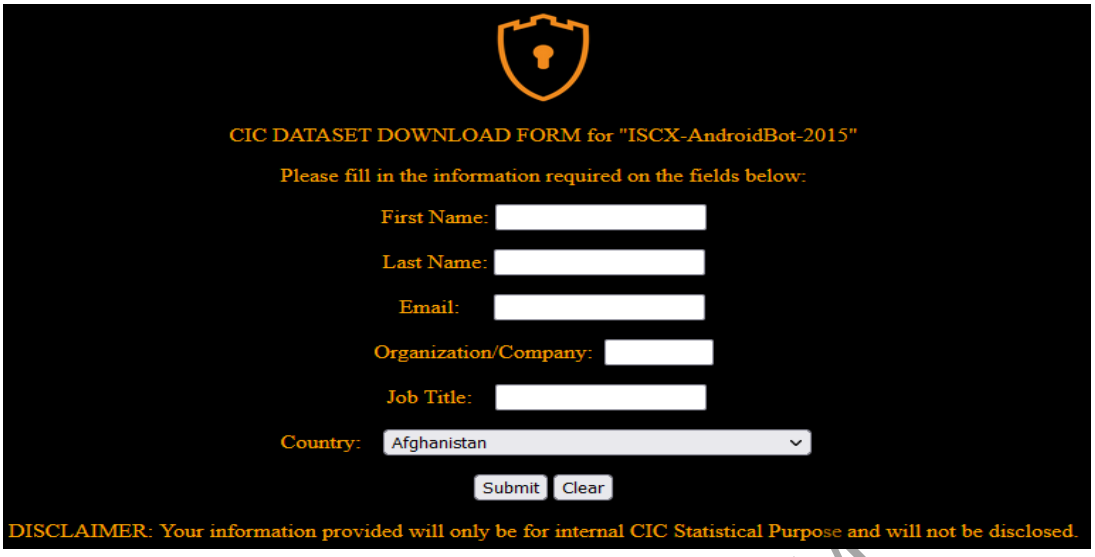
4.2 PENGUMPULAN DAN PEMPROSESAN DATA

Data untuk kajian ini dikumpul daripada laman sesawang pengkaji dari CIC (Android *Botnet* 2015 CIC. (n.d.)) seperti yang dilampirkan dalam **Rajah 4.1** dan **Rajah 4.2**.



Rajah 4.1 Laman sesawang *Canadian Institute for Cybersecurity (CIC)*[13]

[13] https://www.unb.ca/cic/datasets/android-*botnet*.html



Rajah 4.2 Maklumat yang diperlukan untuk memuat turun dataset *Botnet* Android

Data yang dimuat turun mempunyai 1920 ***.apk*** sampel (14 varian) yang ditemui dari tahun 2010 hingga 2014. **Jadual 4.1** melampirkan varian *Android Botnet* beserta dengan jumlah sampel. Terdapat juga set *URLs* yang dikongsi oleh pengkaji Abdul Kadir et al. (2015) berdasarkan hasil kajian mereka pada tahun 2015.

Jadual 4.1 Maklumat set data *botnet* yang dimuat turun

No	Varian (Tahun)	Jumlah Sampel
1	*Anserverbot* (2011)	244
2	Bmaster (2012)	6
3	*Droiddream* (2011)	363
4	*Geinimi* (2010)	264
5	MisoSMS (2013)	100
6	NickySpy (2011)	199
7	Not Compatible (2014)	76
8	*PJapps* (2011)	244
9	Pletor (2014)	85
10	RootSmart (2012)	28
11	Sandroid (2014)	44
12	TigerBot (2012)	96
13	Wroba (2014)	100
14	Zitmo (2010)	80
	Total	**1929**

Dalam fasa ini, set data yang telah dikumpul dibahagikan kepada dua bahagian:

1. **Analisis awal *(pre-liminary)***: Untuk analisis awal, semua data yang berkaitan dengan 14 varian yang telah dibentangkan oleh pengkaji CIC digunakan untuk membuat senarai rumusan semua teknik enkripsi yang boleh digunakan oleh Android *Botnet*. Ini termasuklah teknik-teknik yang biasa digunakan dan juga teknik pengubahsuaian seperti yang dilampirkan dalam kajian mereka.

2. **Pengujian kerangka (*framework evaluation*)**: Untuk peringkat pengujian ini hanya tiga daripada 14 varian telah dipilih iaitu ***Anserverbot*, *Droiddream***, dan ***PJapps* (851 sampel)**. Pemilihan dibuat berdasarkan dapatan yang dilaporkan di dalam kajian literatur di mana ketiga-tiga varian ini sangat aktif dan menggunakan enkripsi di dalam komunikasi *URL* mereka.

4.3 PEMBANGUNAN KERANGKA ANALISIS

Berikut adalah langkah-langkah yang dilakukan dalam fasa pembangunan kerangka ini:

1. **Menetapkan keperluan sistem**: Setiap fasa di dalam kerangka memerlukan keperluan sistem yang berbeza:

2. ***Pengumpul*** memainkan peranan untuk memilih hanya Android (*.apk*) untuk dianalisa ke fasa yang kedua iaitu ***Ekstraktor***.

3. ***Ekstraktor*** pula mempunyai dua modul iaitu statik dan dinamik untuk mendapatkan *URL*s daripada pengumpul (apk). *URL*s ini diperolehi daripada *Strings* yang terdapat di dalam sampel, termasuklah *Strings* yang telah dienkripsi.

4. ***Ejen Analitik*** memerlukan senarai penggunaan enkripsi dan alat yang sedia ada untuk digunakan bagi menyahkod dan menyahsulit sebarang *URL*s. Senarai ini dikumpul dari sumber luar (alat enkrepsi sedia ada) dan hasil kajian literatur dari akademik dan industri.

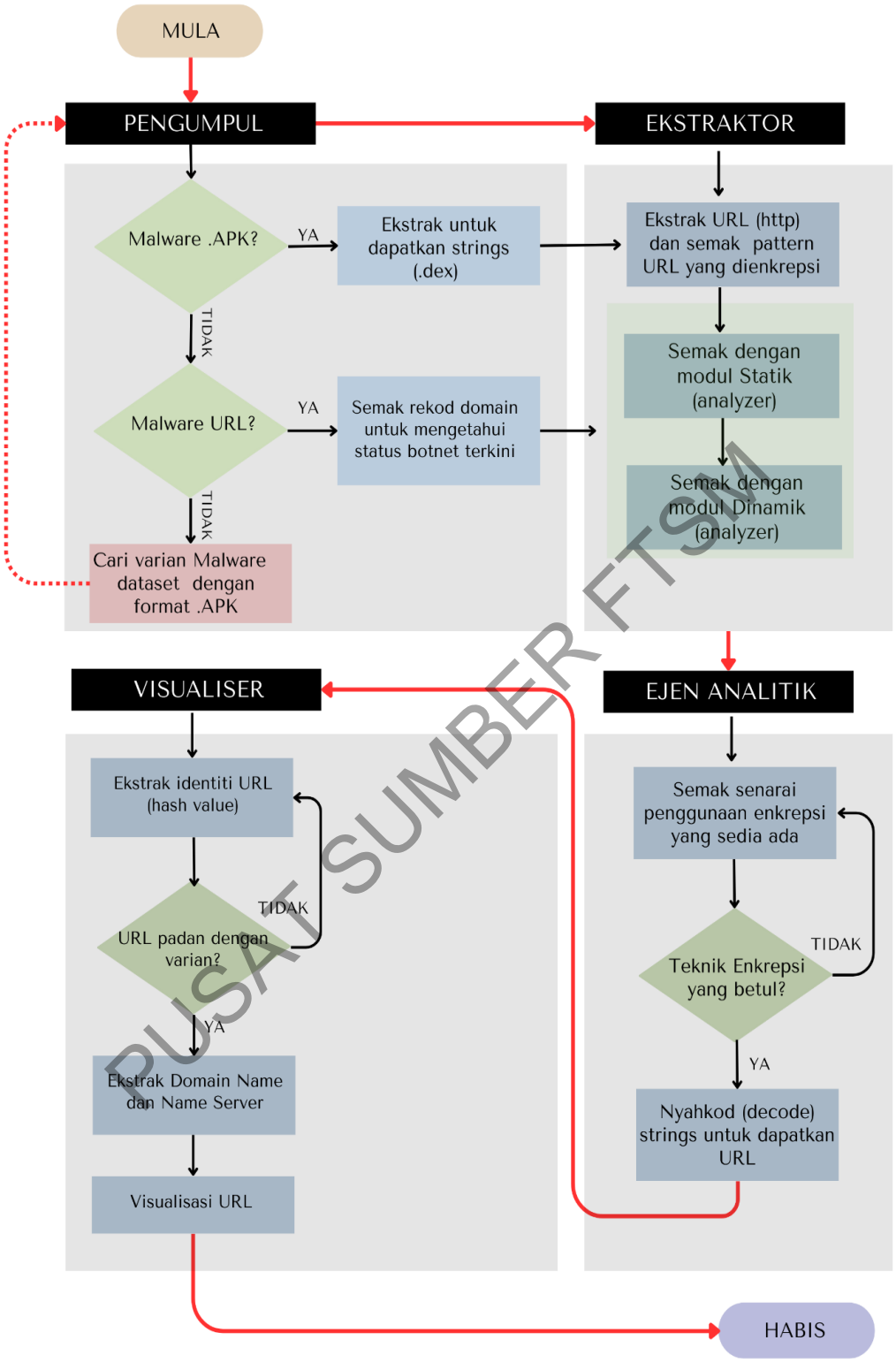
5. ***Visualiser*** memerlukan gabungan penggunaan dua alat visualisasi iaitu *NodeXL* dan *Afterglow* dengan sistem yang berbeza dimana *NodeXL* hanya boleh digunakan di peranti *Windows* sahaja dan *Afterglow* hanya untuk peranti *Linux*.

Rajah 4.3 menunjukkan carta alir bagi keempat-empat fasa ini dari mula pengumpulan data sehingga fasa terakhir iaitu visualisasi.

1. **Merancang reka bentuk**: Untuk memudahkan interaksi dan aliran data keempat-empat komponen kerangka, beberapa sumber luar di terapkan di dalam kerangka ini seperti set data awam, alat enkripsi dan modul analisis yang sedia ada (*Hybrid Analyis, Web-based programming editor)* dapat membantu pengkaji yang mempunyai jumlah data yang kecil untuk dianalisa. Proses ini penting untuk menjimatkan masa analisis.

2. **Pemilihan bahasa pengaturcaraan dan alat visualisasi**: Dalam konteks kerangka analisis untuk *botnet URL*s, *Python* dapat digunakan untuk pengumpulan data, pemprosesan *URL*, dan analisis data.

Dengan kelebihan yang dimiliki *Python*, pengguna dapat mengembangkan skrip yang efisien, fleksibel, dan efektif untuk analisis *botnet URL*s. Untuk alat visualisasi pula, pemilihan dibuat berdasarkan ciri-ciri yang ditawarkan. *NodeXL* lebih fleksibel untuk pengkaji menerokai tingkah laku *botnet* tetapi memerlukan masa yang agak lama untuk penghasilan kerana bergantung kepada kelajuan *Microsoft Excel*. *Afterglow* pula lebih cepat untuk dihasilkan kerana menggunakan terminal Linux, tetapi pemilihan bentuk gambaran adalah terhad tidak seperti *NodeXL*. Gabungan skrip *Python* serta kedua-dua alat visualisasi ini dapat menjadikan kerangka ini lebih fleksibel dan efisien.

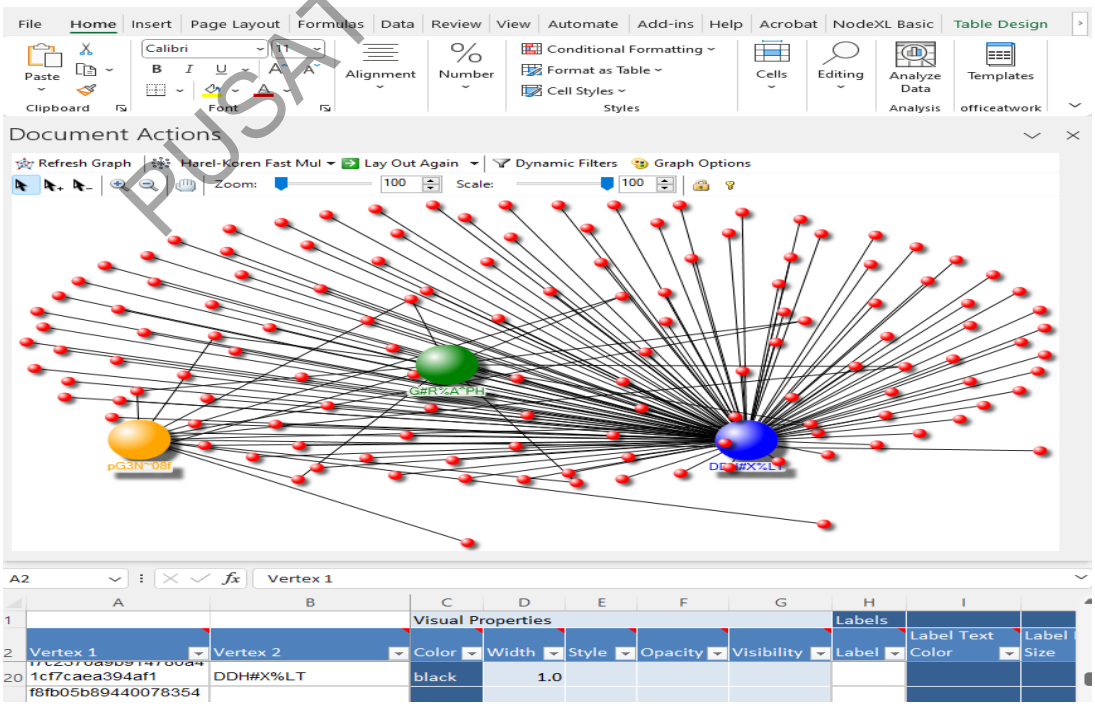


Rajah 4.3 Carta alir dari fasa pengumpul, ekstraktor, ejen analitik, hingga visualiser

Rajah 4.4 dan **Rajah 4.5** menunjukkan contoh paparan bagi alat visualisasi *NodeXL*. Alat ini mempunyai Integrasi yang baik dengan *Microsoft Excel*, membolehkan penggunaan data sedia ada dan analisis statistik.

	A	B	C	D	E	F	G	H	I	J
1			Visual Properties					Labels		
2	Vertex 1	Vertex 2	Color	Width	Style	Opacity	Visibility	Label	Label Text Color	Label Font Size
120	cf7caea394af1	DDH#X%LT	black	1.0						10
121	f8fb05b8944007835454606a54a91e78	DDH#X%LT	black	1.0						10
122	fa101903ee9a58d5fec9ab14448aa585	DDH#X%LT	black	1.0						10
123	fe19e7a0c32e8ef3d1d970cdfa6a1dc2	DDH#X%LT	black	1.0						10
124	ff648ec6c1f22e6147daa7f3fc221892	DDH#X%LT	black	1.0						10
125	2f51e24c3d3fbfeeb9800c8953aa16ed	G#R%A^PH	black	1.0						10
126	5b60b511276103a924f23b64877e1fd8	G#R%A^PH	black	1.0						10
127	5c04d5b12850163d4578269865ad751b	G#R%A^PH	black	1.0						10
128	5d85ceed89326b22dd85f8a39a6f685e	G#R%A^PH	black	1.0						10
129	6f163a2e2fed32cba1a9716b5a608977	G#R%A^PH	black	1.0						10
130	9a3fe8611572b71d2b7a1e73da397bc5	G#R%A^PH	black	1.0						10
131	9bc73e882d565d2d87b503a8e766fe84	G#R%A^PH	black	1.0						10
132	a26687d97b1aa58cd94392fb12dbfb4a	G#R%A^PH	black	1.0						10
133	ced750ec1c97b5c3ab61f4765d21a9fb	G#R%A^PH	black	1.0						10
134	eb2e35189612c5b1801abbd2eae80bb8	G#R%A^PH	black	1.0						10
135	2f51e24c3d3fbfeeb9800c8953aa16ed	pG3N~08f	black	1.0						10
136	58f99ee004c6ac832486e23bc19b0430	pG3N~08f	black	1.0						10
137	5b60b511276103a924f23b64877e1fd8	pG3N~08f	black	1.0						10
138	5c04d5b12850163d4578269865ad751b	pG3N~08f	black	1.0						10

Rajah 4.4 Contoh input data untuk *NodeXL* (teks)

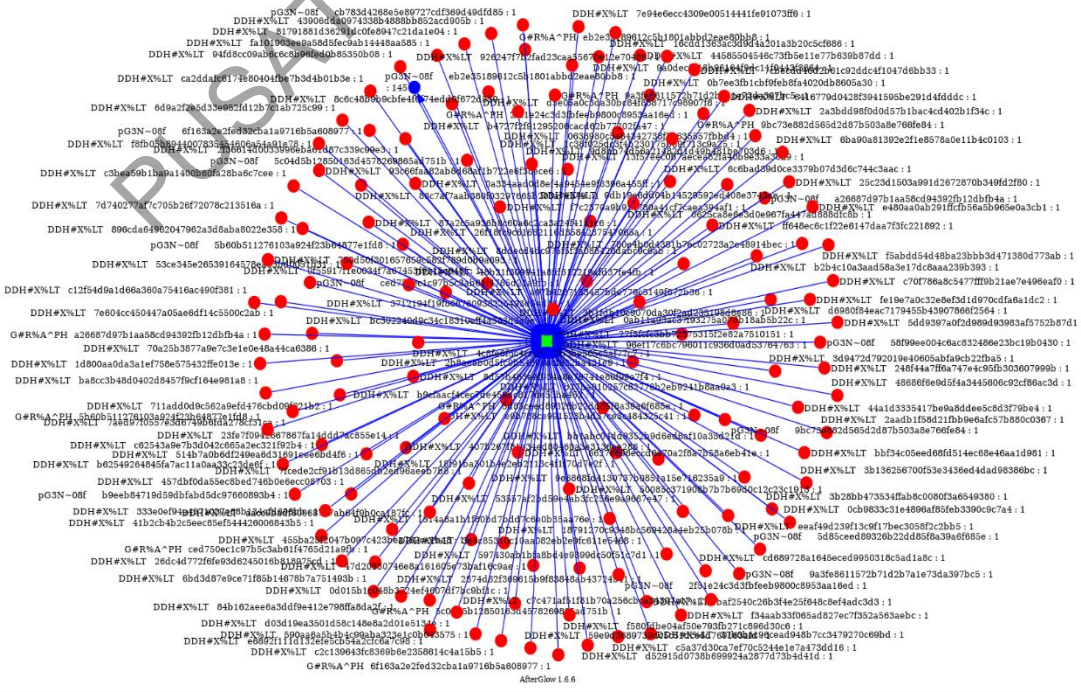


Rajah 4.5 Contoh input data untuk *NodeXL* (teks dan visualisasi)

Rajah 4.6 dan **Rajah 4.7 pula** menunjukkan contoh paparan bagi alat visualisasi *Afterglow*. Alat ini mampu memaparkan jaringan yang besar dengan cara yang lebih efisien. Paparan tetapan untuk alat ini dihasilkan di dalam bahasa pengaturcaraan *Perl.*

```
# AfterGlow Color Property File

# @fields is the array containing the parsed values
# color.source is the color for source nodes
# color.event is the color for event nodes
# color.target is the color for target nodes
#
# The first match wins

color.source="yellow" if ($fields[0]=~/^192\.168\..*/);
color.source="greenyellow" if ($fields[0]=~/^10\..*/);
color.source="lightyellow4" if ($fields[0]=~/^172\.16\..*/);
color.source="red"
color.event="yellow" if ($fields[1]=~/^192\.168\..*/)
color.event="greenyellow" if ($fields[1]=~/^10\..*/)
color.event="lightyellow4" if ($fields[1]=~/^172\.16\..*/)
color.event="green"
color.target="blue" if ($fields[2]<1024)
color.target="lightblue"
color.sourcetarget="pink"

color.edge="blue" if (1)
size.edge=1;

# Changing node labels:
#label=substr(field(),0,10)
#label.edge=$fields[2]       # only for GDF and DOT
#label.duplicate=1       # only for GDF
```

Rajah 4.6 Contoh paparan tetapan untuk *Afterglow*



Rajah 4.7 Contoh paparan untuk output data visualisasi *Afterglow*

Jadual 4.2 menunjukkan kelebihan dan kekurangan alat-alat visualisasi ini.

Jadual 4.2 Perbandingan alat visualisasi

Alat	Kelebihan	Kekurangan
NodeXL	Antara alat visualisasi rangkaian yang popular dan mudah digunakan.	Tidak menyediakan fungsi visualisasi yang sangat kompleks seperti alat lain yang lebih khusus.
	Integrasi yang baik dengan *Excel*, membolehkan penggunaan data sedia ada dan analisis statistik (**Rajah 4.4** dan **Rajah 4.5**).	Terhad dalam hal jenis visualisasi yang boleh dihasilkan, terutamanya untuk visualisasi jaringan yang sangat besar.
	Menyediakan fungsi analisis jaringan yang berguna untuk mengenalpasti struktur dan corak dalam data.	Terbatas dalam kapasiti manipulasi data yang kompleks, terutama apabila berurusan dengan jaringan yang besar.
Afterglow	Membolehkan visualisasi yang kaya dan kreatif dengan fungsi-fungsi yang fleksibel.	Memerlukan pengetahuan teknikal yang mendalam dalam pengaturcaraan untuk menggunakannya sepenuhnya.
	Mampu memvisualisasikan jaringan yang besar dengan cara yang lebih efisien.	Memerlukan sumber daya komputer yang lebih tinggi untuk mengendalikan jaringan yang kompleks dan besar.
	Menyediakan pemboleh ubahan dalam gaya visualisasi dan penyesuaian untuk menghasilkan hasil yang unik.	Kurangnya integrasi dengan alat analisis statistik dan sumber data yang mudah digunakan.

4.4 PENGATURCARAAN DAN ALGORITMA

Skrip *Python* yang digunakan untuk menyahkod dan menyahsulit *URL*s dilampirkan di **Rajah 4.8** dan **Rajah 4.9**. Input *strings* (*URL)* ini diperolehi dari fasa kedua (Ekstrak) dan ketiga (Ejen Analitik) di mana terdapat sampel yang menggunakan corak enkrepsi yang sama dari kajian sebelum ini. Maklumat yang telah dikumpulkan sebelum ini menyatakan beberapa teknik enkripsi yang biasa digunakan. Contohnya pengkaji Abdul Kadir et al. (2015) menyatakan hampir kesemua varian Android *Botnet* menggunakan

enkripsi teknik seperti Base64, cipher rotation (**Rajah 4.10**), XOR, AES, DES, termasuklah pengubahsuaian teknik seperti mengubah jadual indeks bagi Base64 dan menggunakan teknik "*letter skipping*".

Selain itu, terdapat juga hubungan yang signifikan antara *URL* C&C dan corak penyulitan mereka dalam keluarga varian yang sama seperti yang dilamprikan di **Rajah 4.11**. Sebagai contoh, kesemua keluarga *Droiddream* menggunakan teknik yang sama untuk penyulitan *URL* C&C. Keluarga *botnet* ini menyimpan *URL* berkod keras di tempat yang sama dan menentukan kunci untuk menyahsulit *URL* ini dalam kod Java dengan kata kunci `PASSWORD CRYPT KEY` seperti yang dilaporkan oleh McAfee sebelumnya. Jadual 4.3 menunjukkan skrip Linux yang digunakan untuk mengekstrak URL secara pukal dan unik dari *.apk* sampel yang dikumpul.

Penerangan skrip secara terperinci untuk semua skrip Linux dan *Python* berikut dijelaskan di **LAMPIRAN D:** skrip *Linux* untuk mengekstrak *URL* secara pukal, dan **LAMPIRAN E**: skrip *Python* untuk menyahkod *URL PJapps, Droiddream* dan *Anserverbot* varian.

Jadual 4.3 Skrip Linux untuk ekstrak *URL* secara pukal

```
#ekstrak URL dari APK
for i in *.apk; do strings | grep -Eri http > URL.txt "$i"; done


#ekstrak URL dari semua fail (termasuklah .xml, asset, resource)
for p in *; do echo == $p; strings $p| grep -i "http://"; done
#ekstrak URL yang unik
strings * | grep -i "http://" |sed -n '4,$ s@^.*http://\([^/]*\).*$@\1@p' |sed 's/www.//' |sort| uniq –
```

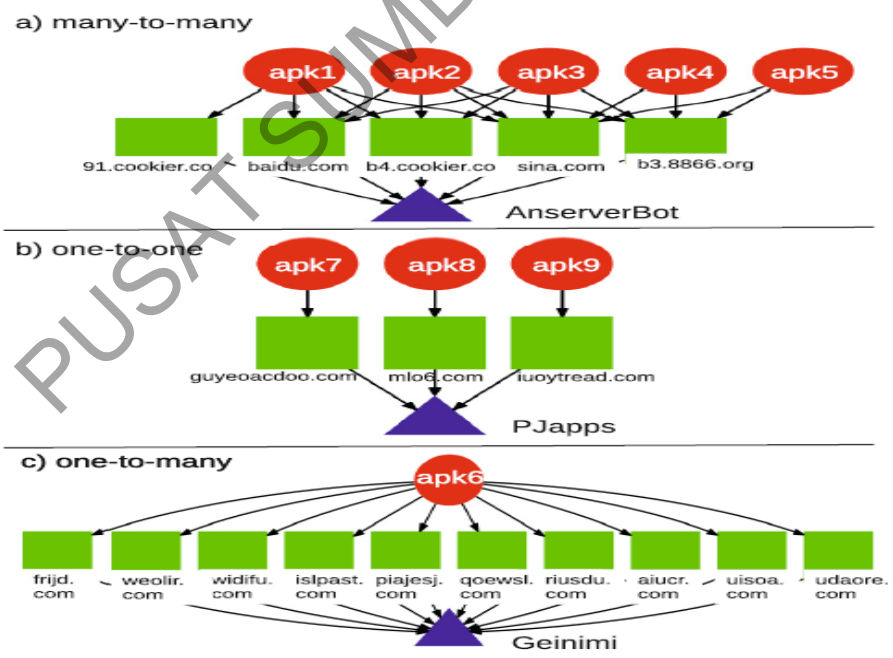
```python
#!/usr/bin/python
#skrip untuk PJapps varian
#@hairul nizam projek master 2023

import string
import sys

s0 =  'kl4ofgsmgeje5gko99s1fc2ofm'
s1='aa3n2d4rdo5i2dspnahoaj3ifa7oxcjn'
s2='acbt32xp2aaogjdixano3cxn'
s3='xga3sg73xcfn'
s4=  '2maodb3ialke8mdeme3gkos9g1icaofm'
s5 ='ax3mkl4mgele2guoo9f1hc3ohm'
s6='31goagdmfejekgfos9t15chojm'

print s0[1:50:2]
print s1[1:50:2]
print s2[1:50:2]
print s3[1:50:2]
print s4[1:50:2]
print s5[1:50:2]
print s6[1:50:2]
```

Rajah 4.8 *Python* skrip untuk menyahkod *URL Pjapps* varian yang diperolehi dari strings yang dienskripsi

```python
#!/usr/bin/python
#skrip untuk Anserverbot varian
#@hairul nizam projek master 2023

import string
import base64
import sys

#input-URL yang dienkrepsi
s0 = 'HoiprJbh9CFE8CrOrCRO7cBw8CpO7CQhr2MW8tMeKNnp0JT57wrQfJjYfoFLPxDOHoig8S__'
s1 = 'HoiprJbh9CFE8CrOrCRO7cBw8CpO7CQhr2MW8tMeKNnp0JT57wrQfJjYfoFOXxyOHoig8S__'
s2 = 'HoiprJbh9CVN9wnQ0w7O84FePwnYPJShHIE29IkwutRh8n__'
s3 = 'HoiprJbh9CVN9wnQ0w7O84FePwnYPJShHIEO7x0pHxMO'
s4 = 'HoiprJbh9CVp9I0h8Cg1zKVO7CAO7CfaPJSQfvMUH2B574i18CQ_'
s5 = 'HoiprJbh9CVp9I0h8Cg1zKVO7CAO7CfaPJSQfvMUHNVO7x0pHxMO'
s6 = 'HoiprJbh9NDs9I0h8Cg1zKVO7CAO7CfaPJSQfvMUHLMwzxBpzKVhPqjh7x1OXtLE'

#jadual indeks yang diubahsuai
my_base64chars = "STvJjktoVFZ9f0PGlicqy3xK7zH8ruXdn5WwDRIeb1UmEgOhYs2NpLC4QBa6AM+/_"

#jadual indeks base64 yang biasa
std_base64chars = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/="

#menterjemah berdasarkan jadual indeks
s0 = s0.translate(string.maketrans(my_base64chars, std_base64chars))
s1 = s1.translate(string.maketrans(my_base64chars, std_base64chars))
s2 = s2.translate(string.maketrans(my_base64chars, std_base64chars))
s3 = s3.translate(string.maketrans(my_base64chars, std_base64chars))
s4 = s4.translate(string.maketrans(my_base64chars, std_base64chars))
s5 = s5.translate(string.maketrans(my_base64chars, std_base64chars))
s6 = s6.translate(string.maketrans(my_base64chars, std_base64chars))

#menyahkod
data0 = base64.b64decode(s0)
data1 = base64.b64decode(s1)
data2 = base64.b64decode(s2)
data3 = base64.b64decode(s3)
data4 = base64.b64decode(s4)
data5 = base64.b64decode(s5)
data6 = base64.b64decode(s6)

#output
print data0
print data1
print data2
print data3
print data4
print data5
print data6
```

Rajah 4.9 *Python* skrip untuk menyahkod *URL Anserverbot* varian yang diperolehi dari strings yang dienkripsi

Rajah 4.10 Alat yang digunakan untuk menyahkod *URL Droiddream* Varian yang dienkripsi[14]



Rajah 4.11 Jenis hubungan komunikasi *Botnet* mengikut varian

Sumber: Abdul Kadir et al. 2015

[14] https://rot13.com/

4.5 KESIMPULAN

Bab ini telah memberikan gambaran secara keseluruhan mengenai pelaksanaan kajian yang dilakukan. Pada bahagian pengenalan, tujuan dan latar belakang kajian ini telah diberikan. Seterusnya, dalam bahagian pengumpulan dan pemprosesan data, data yang relevan telah dikumpulkan dan diproses dengan menggunakan kaedah yang sesuai untuk analisis. Proses pembangunan kerangka analisis kemudiannya dilakukan untuk membentuk struktur dan landasan bagi analisis *botnet* Android *URL*s yang dienkripsi. Dalam bahagian pengaturcaraan dan algoritma, penulisan kod dan penggunaan algoritma telah dilakukan untuk melaksanakan proses analisis dengan cara yang lebih teknikal. Maklumat dan langkah-langkah yang diperoleh dari pelaksanaan kajian ini akan menjadi asas untuk meneruskan kepada bahagian-bahagian seterusnya dalam kajian ini, dan membantu dalam pencapaian objektif kajian secara keseluruhan.

BAB V

DAPATAN KAJIAN

5.1 PENGENALAN

Bab ini membentangkan dapatan kajian hasil dari pengujian dan penilaian kerangka analisis yang dibahagikan kepada dua sub-topik: dapatan kajian penggunaan enkripsi dan penggunaan visualisasi. Keseluruhannya, terdapat 851 sampel apk dari tiga varian (*Anserverbot*, *Droiddream*, *PJapps*) yang dianalisa untuk pengujian dan penilaian ini.

5.2 ANALISA PENGGUNAAN ENKRIPSI

Secara kesimpulannya, dapatan kajian menunjukkan kerangka analisis telah berjaya menyahsulit (menggunakan 3 set kunci) dan menyahkod **32 *URL*** unik dari 851 sampel apk; 8 *URL* dari varian *Anserverbot*, 17 *URL* dari *Droiddream*, dan 7 *URL* dari varian *PJapps* (rujuk **Jadual 5.1**). Paparan yang terperinci untuk setiap *URL* ini ada dilampirkan di **Jadual 5.2**, **Jadual 5.3, Jadual 5.4, Jadual 5.5.**

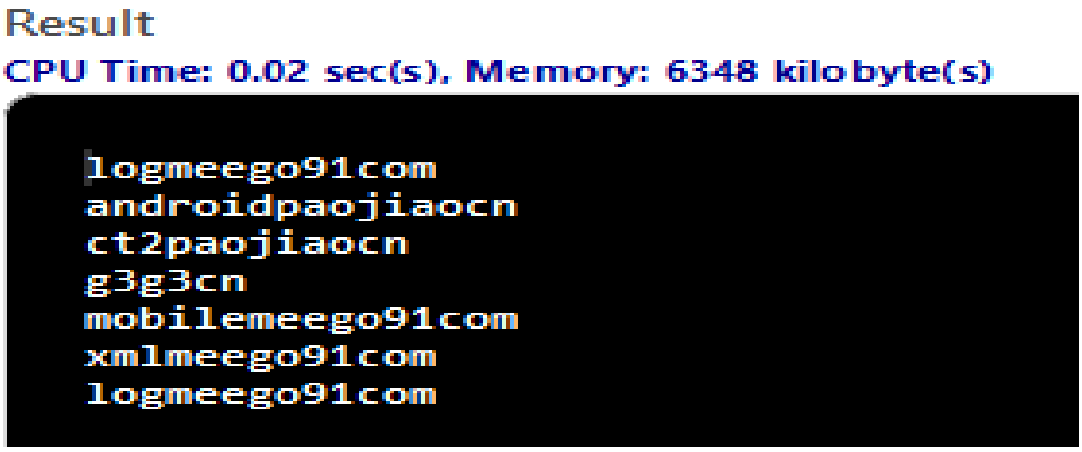
Jadual 5.1 Jumlah *URL* yang unik mengikut varian

Varian	**Teknik Enkripsi**	**Jumlah Sampel**	**Jumlah *URL*s (Unik)**
Anserverbot	Base64 (pengubahsuaian jadual indeks)	244	8
Droiddream	Rotation Cipher (Rot-16)	363	10
	AES (dengan 3 set kunci)		7
PJapps	*Skipping a letter*	244	7
	Jumlah	**851**	**32**

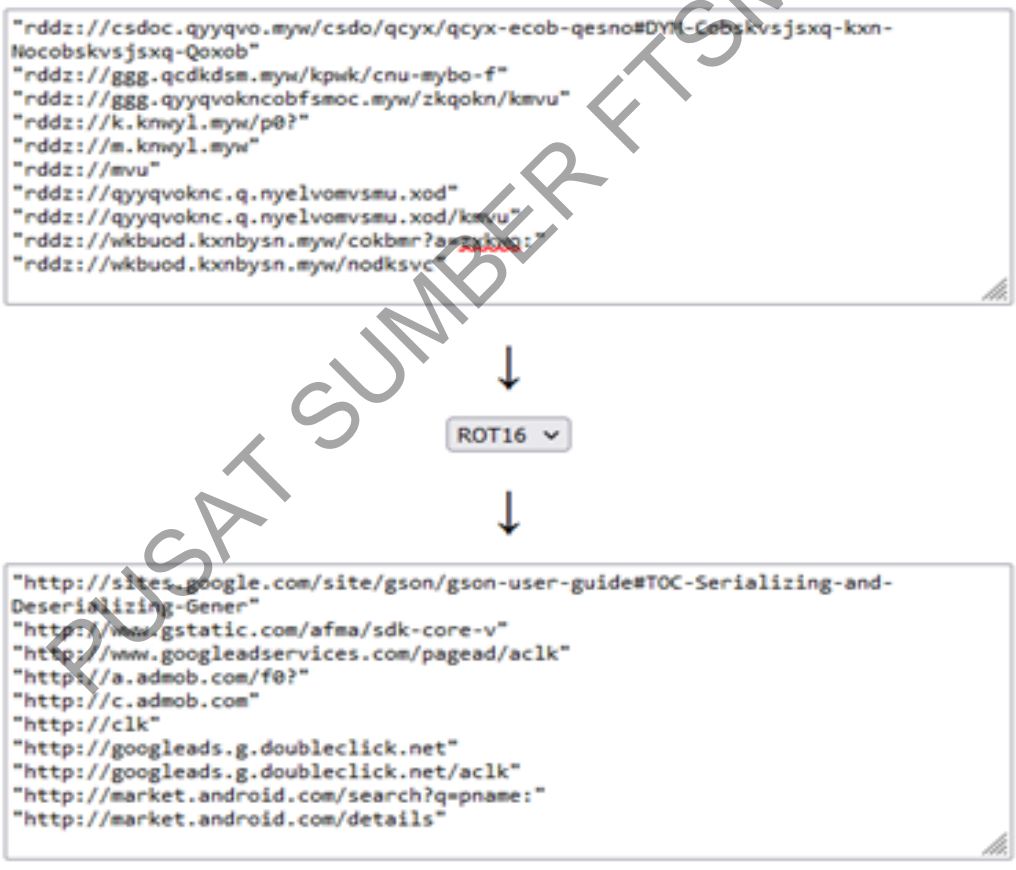
Maklumat yang telah dikumpulkan dari hasil kajian literatur ada menyatakan beberapa teknik enkripsi yang biasa digunakan. Contohnya kajian dari Chen, H., & Zhou, Y. (2013), Abdul Kadir et al. (2015), dan Alqatawna, J. F., & Faris, H. (2017, October) ada menyatakan hampir kesemua varian Android *Botnet* menggunakan enkripsi teknik seperti Base64, cipher rotation, XOR, AES, DES, termasuklah pengubahsuaian teknik seperti mengubah jadual indeks bagi Base64 dan menggunakan teknik "*letter skipping*". Maklumat dari *VirusTotal*. (n.d.) juga sangat membantu dalam penemuan dapatan kajian untuk ketiga-tiga varian ini (*Anserverbot*, *Droiddream*. *PJapps*).

Selain itu, penggunaan kata kunci di dalam pencarian *strings* juga dapat membantu dalam pengumpulan *URLs*. Contohnya, pencarian kata kunci *"password", "AES", "DES"* membantu dalam mengenalpasti URL yang menggunakan teknik AES dengan 3 set kata laluan (*pG3N~08f, G#R%A^PH, DDH#X%LT*) seperti yang dilampirkan dalam **Jadual 5.4**. Ketiga-tiga kata laluan ini dikenalpasti melalui pencarian strings di lokasi *asset file* yang dinamakan sebagai */assets/prefer.dat.* Penggunaan kata kunci di sini adalah untuk tujuan analisis dan pengumpulan *URL* yang berkaitan dengan enkripsi, bukan untuk mengkaitkan dengan *salting AES* atau enkripsi data dengan *salt*. **Rajah 5.1**, **Rajah 5.2**, dan **Rajah 5.3** menunjukkan contoh output *URL* yang telah dinyahsulitkan mengikut varian masing-masing.

Rajah 5.1 Contoh output *URL* yang telah dinyahsulit dari varian *Anserverbot*



Rajah 5.2 Contoh output *URL* yang telah dinyahsulit dari varian *PJapps*



Rajah 5.3 Contoh output *URL* yang telah dinyahsulit dari varian *Droiddream*

Jadual 5.2 Statistik *URL* untuk Varian *Anserverbot* yang berjaya dinyahkod (Teknik pengubahsuaian Base 64)

No	***URL*s yang dienkripsi**	***URL*s yang telah dinyahkod**	**Jumlah**
1	HoiprJbh9CFE8CrOrCRO7cBw8CpO7CQhr2MW8tMeKNnp0JT57wrQfJjYfoFLPxDOHoig8S__	http://blog.sina.com.cn/s/blog_8440ab780100ru9i.html	175
2	HoiprJbh9CFE8CrOrCRO7cBw8CpO7CQhr2MW8tMeKNnp0JT57wrQfJjYfoFOXxyOHoig8S__	http://blog.sina.com.cn/s/blog_8440ab780100rnye.html	231
3	HoiprJbh9CVN9wnQ0w7O84FePwnYPJShHIE29IkwutRh8n__	http://b3.8866.org:8080/jk2.action	7
4	HoiprJbh9CVN9wnQ0w7O84FePwnYPJShHIEO7x0pHxMO	http://b3.8866.org:8080/jk.action	133
5	HoiprJbh9CVp9I0h8Cg1zKVO7CAO7CfaPJSQfvMUH2B574i18CQ_	http://b4.cookier.co.cc:8080/jk.action	98
6	HoiprJbh9CVp9I0h8Cg1zKVO7CAO7CfaPJSQfvMUHNVO7x0pHxMO	http://b4.cookier.co.cc:8080/jk2.action	2
7	HoiprJbh9NDs9I0h8Cg1zKVO7CAO7CfaPJSQfvMUHLMwzxBpzKVhPqjh7xlOXtLE	http://91.cookier.co.cc:8080/jk_center/91/ad.xml	175
8	HoiprJbh9C519IF5HxiL9I0h8cMNuezDrebh7Ishz2M1ut3g9C7CPt05zI7s0xlQzwTRfNlpzqi5zxjL0cBbutLE	http://hi.baidu.com/svvdrz/blog/item/f68caff15d8f0e344e4aea55.html	
		Jumlah	**821**

Jadual 5.3 Statistik *URL* untuk Varian *PJapps* yang berjaya dinyahkod (Teknik *Skipping a letter*)

No	***URL*s yang dienkripsi**	***URL*s yang telah dinyahkod**	**Jumlah**
1	http://kl4ofgsmgeje5gko99s1fc2ofm	http://logmeego91com	65
2	http://aa3n2d4rdo5i2dspnahoaj3ifa7oxcjn	http://androidpaojiaocn	49
3	http://acbt32xp2aaogjdixano3cxn	http://ct2paojiaocn	49
4	http://xga3sg73xcfn	http://g3g3cn	49
5	http://2maodb3ialke8mdeme3gkos9g1icaofm	http://mobilemeego91com	65
6	http://ax3mkl4mgele2guoo9f1hc3ohm	http://xmlmeego91com	65
7	http://3lgoagdmfejekgfos9t15chojm	http://logmeego91com	65
		Jumlah	**407**

Jadual 5.4 Statistik *URL* untuk Varian *Droiddream* yang berjaya dinyahkod (Teknik AES dengan 3 set kata laluan)

No	Kata kunci AES yang digunakan	*URL*s yang telah dinyahsulit (menggunakan kunci AES)	Jumlah
1	pG3N~08f G#R%A^PH DDH#X%LT	http://guyeoacdoo.com/wmzq.jsp http://iuoytread.com	25
2		http://ju5o.com/zpmq.jsp',	1
3		http://iuoytread.com	28
4		http://juodaleety.com/vrzl.jsp	26
5		http://mlo6.com/owxnf.jsp',	1
6		http://oucameyed.com	28
7		http://ya3k.com/bksy.jsp',	1
		Jumlah	**110**

Jadual 5.5 Statistik *URL* untuk Varian *Droiddream* yang berjaya dinyahkod (Teknik Rot-16-Cipher)

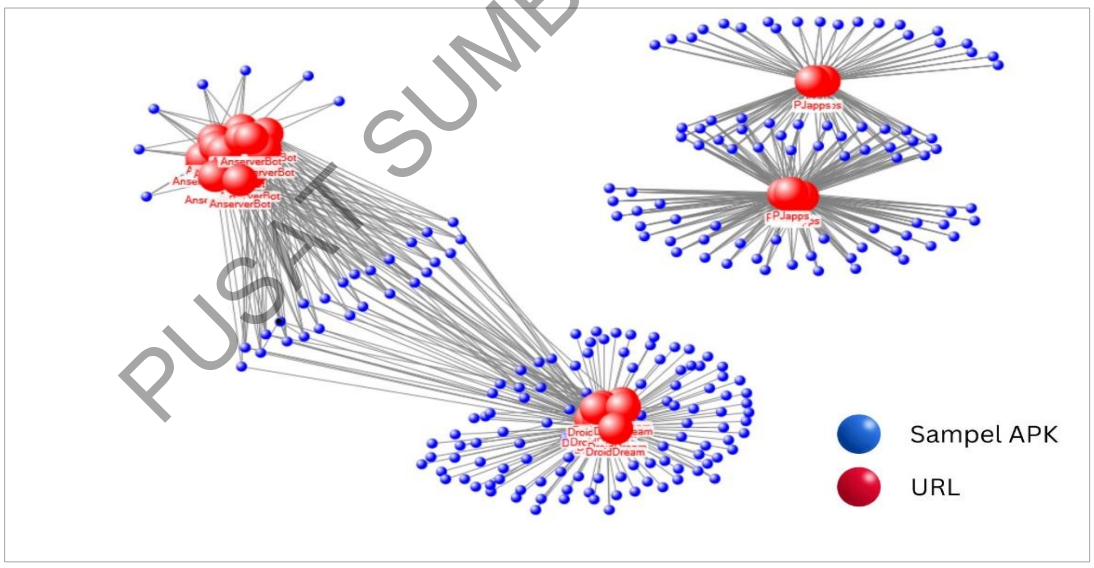
No	*URL*s yang dienkripsi	*URL*s yang telah dinyahkod	Jumlah
1	rddz://csdoc.qyyqvo.myw/csdo/qcyx/qcyx-ecob-qesno#DYM-Cobskvsjsxq-kxn-Nocobskvsjsxq-Qoxob	http://sites.google.com/site/gson/gson-user-guide#TOC-Serializing-and-Deserializing-Gener	15
2	rddz://ggg.qcdkdsm.myw/kpwk/cnu-mybo-f	http://www.gstatic.com/afma/sdk-core-v	15
3	rddz://ggg.qyyqvoknсobfsmoc.myw/zkqokn/kmvu	http://www.googleadservices.com/pagead/aclk	16
4	rddz://k.knwyl.myw/p0?	http://a.admob.com/f0?	16
5	rddz://m.knwyl.myw	http://c.admob.com	16
6	rddz://mvu	http://clk	16
7	rddz://qyyqvoknc.q.nyelvomvsmu.xod	http://googleads.g.doubleclick.net	32
8	rddz://qyyqvoknc.q.nyelvomvsmu.xod/kmvu	http://googleads.g.doubleclick.net/aclk	32
9	rddz://wkbuod.kxnbysn.myw/cokbmr?a=zxkwo:	http://market.android.com/search?q=pname:	2
10	rddz://wkbuod.kxnbysn.myw/nodksvc	http://market.android.com/details	31
		Jumlah	**191**

5.3 ANALISA PENGGUNAAN VISUALISASI

Penggunaan visualisasi dapat meningkatkan pemahaman pengkaji berkenaan tingkahlaku setiap *botnet* varian dalam menggunakan teknik enkripsi pada *URL*.

5.3.1 Varian Berkongsi *URL* yang Sama (Indikasi Penggunaan Teknik *Repackaging*)

Visualisasi ketiga-tiga varian (32 *URL*) menunjukkan bahawa varian ini berkongsi sumber *URL* yang sama dalam komunikasi *botnet* mereka seperti yang digambarkan dalam **Rajah 5.4.** Bulatan besar (merah) adalah *URL* manakala bulatan kecil (biru) adalah kesemua 851 sampel apk**.** Dapat dilihat dengan jelas varian *Anserverbot* dan *Droiddream* mempunyai hubungan *many-to-many* dimana terdapat kluster sampel yang berkongsi *URL* yang sama. Secara kontranya, *PJapps* tidak mempunyai hubungan dengan varian lain tetapi masih berkongsi secara internal.



Rajah 5.4 Hubungan antara sampel *apk* dengan *URLs* menunjukkan perkongsian URL yang sama (indikasi teknik *repackaging*)

Hal ini memberikan gambaran bahawa sampel ini menggunakan teknik *repackaging* dalam penghasilan malware kerana terdapat persamaan antara varian dari segi tingkahlaku dan komunikasi di mana varian ini tidak menghasilkan *malware* yang unik tetapi hanya menggunakan teknik *repackaging* (salinan semula) untuk menjimatkan kos penghasilan *malware.*