CyberSecurity dan Forensik Digital e-ISSN: 2615-8442
Vol. 5, No. 2, November 2022, hlm. 71-76

ANALISIS LALU LINTAS JARINGAN TERENKRIPSI DARI *SECURE INSTANT MESSAGING APPLICATION:* STUDI KASUS PADA APLIKASI PESAN INSTAN SYNOLOGY CHAT

Grinaldy Yafi' Rasyad[1], Dedy Hariyadi[2], Tri Febrianto[3]

[1] Universitas Gadjah Mada
[2] Universitas Jenderal Achmad Yani Yogyakarta
[3] PT Widya Adijaya Nusantara
Email: [1] grinaldy.yafi@mail.ugm.ac.id, [2] dedy@unjaya.ac.id, [3] tri.febrianto@widyasecurity.com

Abstrak

Aplikasi pesan instan mempunyai peran yang sangat penting dalam interaksi secara daring terutama saat kondisi pandemi COVID-19. Aplikasi pesan instan yang aman dan terpercaya bagi sebagian orang memberikan kenyamanan dan manfaat. Tulisan ini mengkaji tentang eksperimen forensik jaringan berupa analisis lalu lintas jaringan terenkripsi dari aplikasi pesan instan Synology Chat yang diidentifikasi pola lalu lintas jaringan komunikasi menggunakan aplikasi Wireshark. Pengujian pada penelitian ini dilakukan dengan eksperimen secara langsung menggunakan infrastruktur yang selaras dengan metode uji *Man In the Middle*, yaitu pemantauan lalu lintas jaringan pada komunikasi *client* dan *server*. Analisis dilakukan dengan memantau pola jaringan berupa alamat IP dan besar paket yang dikirimkan dan protokol yang digunakan. Pemantauan pola jaringan dilakukan pada tiga *channel* yaitu *public group*, *private group*, dan *personal message*. Berdasarkan pembuktian bahwa pola besar paket yang dikirimkan *public group* memiliki pola yang hampir sama dengan *personal chat* dan *private group* memiliki pola besar paket yang lebih besar daripada *channel* yang lain. Protokol dominan menggunakan TLSv1.2 sebagai protokol pengiriman data yang terenkripsi. Berdasarkan temuan tersebut pola dapat digunakan sebagai petunjuk jika hendak melakukan investigasi forensik pada kasus tertentu yang menggunakan aplikasi pesan instan Synology Chat.

Kata kunci: synology chat, wireshark, forensik jaringan, pesan instan, MITM.

Analysis of Encrypted Network Traffic from a Secure Instant Messaging App: A Case Study on a Synology Instant Messaging App

Abstract

Instant messaging applications have a very important role in online interactions, especially during the COVID-19 pandemic. A safe and reliable instant messaging application for some people provides convenience and benefits. This paper examines network forensic experiments in the form of analysis of encrypted network traffic from the Synology Chat instant messaging application which identifies communication network traffic patterns using the Wireshark application. Tests in this study were carried out by direct experiments using infrastructure that is in line with the Man In the Middle test method, namely monitoring network traffic on client and server communications. Analysis is carried out by monitoring network patterns in the form of IP addresses and the size of packets sent and the protocol used. Network pattern monitoring is carried out on three channels, namely public group, private group, and personal message. Based on evidence that the packet size pattern sent by the public group has a pattern that is almost the same as personal chat and the private group has a larger packet size pattern than other channels. The dominant protocol uses TLSv1.2 as the protocol for sending encrypted data. Based on these findings, the pattern can be used as a guide if you want to carry out a forensic investigation on certain cases using the Synology Chat instant messaging application.

Keywords*: synology chat, wireshark, network forensic, instant messaging, MITM*

1. PENDAHULUAN

Permasalahan yang menjadi perhatian di tengah masyarakat yaitu keamanan privasi pengguna suatu aplikasi. Masyarakat khawatir jika aplikasi yang digunakan tidak aman karena ada pihak ketiga yang mencoba memantau aktifitas pengguna. Salah satu aplikasi yang digunakan di masyarakat yaitu aplikasi pesan instan daring. Pemilihan aplikasi pesan instan daring ini karena banyak memberikan keuntungan terhadap fitur yang ditawarkan seperti kenyamanan dalam menggunakan, kecepatan, dan fitur yang lengkap sesuai dengan kebutuhan pengguna. Pada tahun 2020 saat pandemi COVID-19 pesan instan banyak digunakan sebanyak 4.3 triliun dengan kenaikan sebesar 9% didukung oleh pekerjaan yang

dominan dikerjakan di rumah atau *work from home* (Afzal et al., 2021). Seperti yang diketahui pesan instan yang sering digunakan yaitu WhatsApp, Facebook Messenger, Telegram, LINE dan masih banyak lagi, mereka mengusahakan kenyamanan pengguna untuk menggunakan aplikasi mereka seperti menjaga privasi data dengan menggunakan teknik enkripsi dalam mengirim data. Kualitas pada aplikasi ini juga membantu perusahaan untuk mencari peluang, pemasaran dan periklanan yang ditawarkan oleh platform tersebut. Keamanan dan kenyamanan pengguna tentunya membutuhkan waktu untuk mengembangkannya dan pengembang tentunya tidak luput dari kekurangan dalam mengembangkan aplikasi sehingga memungkinkan orang yang tidak bertanggung jawab dapat memanfaatkan kerentanan pada aplikasi tersebut. Penelitian ini mempunyai tujuan untuk mempraktikkan aktifitas berkirim pesan dan menganalisis pola lalu lintas jaringan pada pesan instan Synology Chat yang terenkripsi. Synology Chat merupakan fitur dari *Network Attach Storage* (NAS) merk Synology yang berfungsi sebagai *chat server*. NAS Synology selain sebagai server berbagi berkas memiliki banyak fitur diantaranya sebagai *chat server* (Sidik & Putra, 2018). Dari penelitian ini dapat memberikan manfaat dan kontribusi, yaitu membantu mengidentifikasi pola lalu lintas jaringan terenkripsi pada pesan instan Synology Chat serta memastikan bahwa pesan instan Synology Chat ini aman digunakan berdasarkan temuan yang didapatkan.

2. METODE PENELITIAN

Jumlah pengguna yang banyak dan popularitas pada suatu media sosial terutama pesan instan daring menjadi suatu bahan penelitian yang menarik dari sudut pandang forensik jaringan. Sebagai contoh beberapa penelitian yang berkaitan dengan lalu lintas jaringan yang dilakukan oleh Walnycky, dkk melakukan menganalisis perangkat dan lalu lintas jaringan pada 20 aplikasi dengan proses mengoleksi bukti berupa *password*, gambar dan lain-lain. Hal serupa juga dilakukan oleh Yusoff, dkk yaitu berupa analisis forensik jaringan pada sosial media yang terkenal seperti Facebook, Twitter dan Telegram pada Firefox *Mobile Simulator* adapun simulator tersebut dipantau jaringan komunikasinya menggunakan Wireshark kemudian berhasil didapatkan IP *Address*, *port, domain* dan *subdomain* tetapi tidak dapat mengidentifikasi pola yang berbeda pada aktifitas tertentu. Penelitian lain yang serupa berhasil mengidentifikasi pola lalu lintas jaringan terenkripsi pada aktifitas tertentu tetapi terbatas pada IMO app (Sudozai et al., 2018). Pada penelitian lain lagi yang dilakukan oleh Conti et al. aplikasi seprti seperti Facebook, Gmail, dan Twitter dilakukan *decode* pada aksi dari *user* menggunakan *machine learning* berdasarkan *domain filtering, packet filtering, packet intervals,* dan *timeout.* Hasil dari penelitian tersebut dapat membantu dalam mempelajari aktifitas *user* atau lebih, dengan demikian dalam membantu membuat keputusan oleh ahli kemanan siber. Penelitian dari Università del Piemonte Orientale melalukukan penelitian tentang aplikasi automasi AnForA bahwa aplikasi android yang terinstal pada perangkat android virtual dapat dilakukan pemantauan serangkaian aktifitas pada penyimpanan perangkat (Anglano et al., 2020). Penelitian lain yang dilakukan oleh Rathi et al. bahwa hasil dari analisis aplikasi WeChat, Viber, WhatsApp dan Telegram tersebut menunjukkan adanya kemungkinan dapat mengambil file database yang terenskripsi pada aplikasi yang terinstall pada perangkat yang *rooted.* Penelitian lain lagi yang berkaitan dengan analisis jaringan yang mempunyai fokus pada aplikasi WhatsApp ketika menelpon dengan men-*decrypt* lalu lintas jaringan (Karpisek et al., 2015). Pada penelitian lain terdapat analisis forensik pada sosial media seperti Facebook, Twitter dan LinkedIn pada *platform* yang berbeda seperti iOS, Android, Windows, dan Blackberry dengan hasil bahwa *platform* Blackberry tidak dapat memulihkan memori perangkat seperti lokasi, basis data dengan informasi tentang *visited profiles, names, tweets, contacts, profile ID* (Awan., 2015).

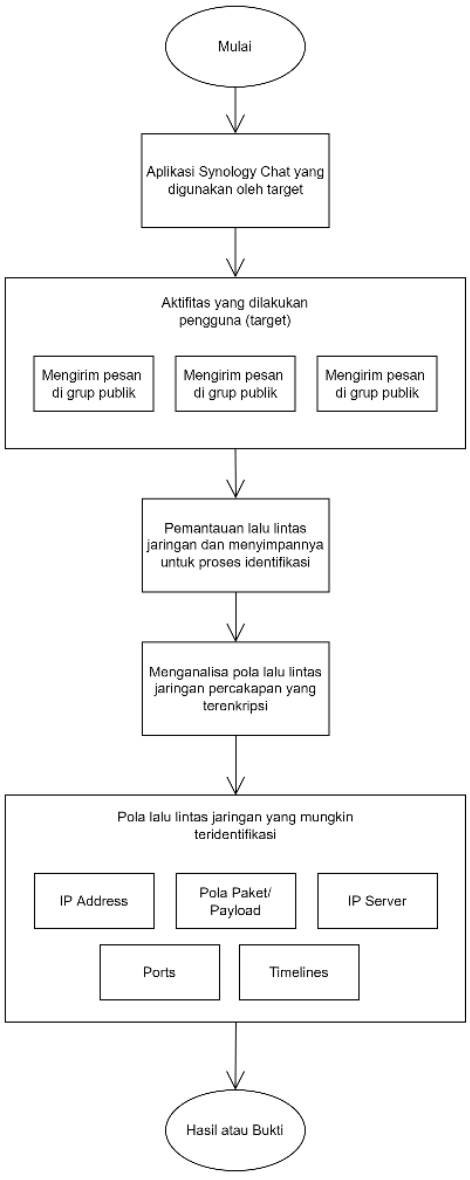
Penelitian ini dilakukan dengan praktik secara langsung berupa eksperimen pada infrastruktur yang sudah diatur sesuai dengan metode *Man In the Middle* yaitu menempatkan pihak ketiga untuk melakukan pemantauan lalu lintas jaringan percakapan antara *client* dan *server* (Mallik, 2018). Penelitian dimulai dengan mengaktifkan alat pemantuan menggunakan Wireshark yang kemudian pengguna mengirim pesan pada *channel public group, private group* dan *personal chat*yang kemudian hasil dari pemantauan tersebut dianalisis polanya yang dapat berupa IP *server dan client, ports, timeline* dan pola *payload* atau paket yang dikirimkan, seperti pada gambar 1

Pada Tabel 1 memberikan informasi bahwa alat yang digunakan untuk eksperimen ini memiliki fungsinya masing-masing terutama untuk menganalisis suatu pola pada pesan yang dikirimkan, bagaimana *client* dan *server* berkomunikasi dan menggunakan protokol dan *ports* apa saja, akan sulit jika alat yang digunakan tidak tepat.

Tabel 1 Peralatan Penelitian

Perangkat	**Tujuan**	**Merek**	***Software/OS***
Router ZTE	LAN Connection	ZTE	V7.0.10P1N14
Kabel LAN	Wired Connection	-	-
Desktop Computer	Deploy Wireshark	Lenovo	Windows 10, 64- bit dengan x64-based processor, 8 GB RAM, Intel (R) Core (TM) i7-5500U CPU @2.40 GHz.

Perangkat	Tujuan	Merek	*Software/OS*
Wireshark	Membaca lalu lintas jaringan	-	3.6.0
Aplikasi Synology Chat	Melakukan aktifitas kirim pesan	Synology Inc	2.7.1
Smartphone	Melakukan aktifitas di aplikasi Synology Chat	Xiaomi POCO X3	Android 12



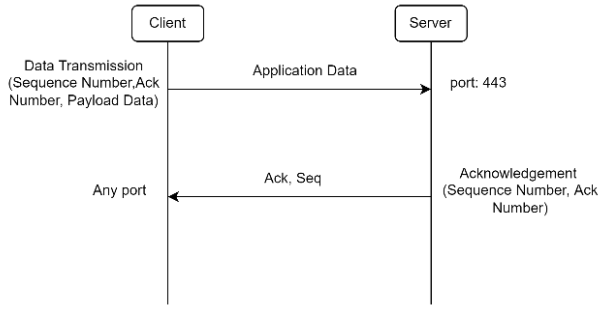
Gambar 1. Alur Analisis dan Pemantauan

Agar mesin pemantau dapat mencatat lalu lintas jaringan komunikasi, diatur infrastruktur yang berupa jaringan internet yang disediakan oleh ISP (*internet service provider)* menggunakan router yang kemudian disambungkan menggunakan kabel LAN yang selanjutnya laptop sebagai mesin pemantau tersebut diatur sebagai *wireless access point*, seperti pada gambar 2. Ponsel yang menajalankan aplikasi Synology Chat akan terhubung ke laptop yang sudah dikonfigurasikan sebagai *wireless access point*. Skenario dimulai ketika pengguna mengirimkan pesan kemudian dipantau lalu lintas jaringannya oleh laptop menggunakan aplikasi Wireshark. Metode ini biasa disebut *Man In The Middle* yang berfungsi untuk memantau aktivitas target saat berkomunikasi, pada penelitian ini komunikasi dengan server Synology Chat.

Gambar 2. Skenario Pemantauan Lalu Lintas Jaringan Komunikasi Synology Chat

Skenario dilakukan dengan memantau lalu lintas yang terjadi antara pengguna (*target*) dengan *server*. Dari skenario tersebut didapatkan lalu lintas jaringan komunikasi yang ditangkap menggunakan aplikasi Wireshark yang kemudian dianalisis pola lalu lintas jaringan berkirim pesan pada aplikasi Synology Chat. Proses pengiriman pesan pada aplikasi Synology Chat melalui proses TCP *Data Transfer* yang ditunjukkan pada gambar 3 dimulai ketika pengguna mengirim pesan (*client*) yang kemudian *application data* atau pesan yang akan dikirimkan ditransfer ke *server* pada *port* 443 dengan protokol keamanan TLSv1.2. Kemudian *server* membalas apa yang sudah dikirimkan oleh *client* berupa *acknowledgement* yang berisi *sequent number*, *Ack number* yang kemudian dikirim kan ke *port* yang dituju (*client*).

Pengiriman pesan dilakukan di tiga *channel* yang berbeda, yaitu *public group*, *private group* dan *personal message*. Pada tiga *channel* tersebut dikirimkan pesan yang sama yaitu "TESTING CHAT".



Gambar 3. TCP Data Transfer pada Proses Pengiriman Pesan

1. Public Group

Public group merupakan grup yang memungkinkan semua orang dapat melihat dan semua dapat bergabung tanpa undangan (Inc., 2022). Percobaan pertama yaitu dengan mengirimkan pesan "TESTING CHAT" pada *public group* untuk mengetahui pola setiap pesan jika dikirimkan melalui *channel public group* kemudian setiap pesan dicatat pola lalu lintas jaringannya menggunakan Wireshark. Gambar 4 menunjukan lalu lintas jaringan berkirim pesan yang tercatat menggunakan Wireshark. Pesan yang terkirim pertama kali di *public group* yang tercata pada waktu ke 0 memberikan informasi bahwa data terkirim dari IP *client* (192.168.137.31) mengirimkan *Application Data* dengan paket sebesar 185 yang dikirimkan melalui protokol TLSv1.2 ke tujuan IP *server* (103.142.110.26).

	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.137.31	103.142.110.26	TLSv1.2	185	Application Data
2	0.056606	103.142.110.26	192.168.137.31	TCP	1494	443 → 44652 [ACK] Se
3	0.056717	103.142.110.26	192.168.137.31	TCP	1494	443 → 44652 [ACK] Se
4	0.056777	103.142.110.26	192.168.137.31	TCP	1494	443 → 44652 [ACK] Se
5	0.056823	103.142.110.26	192.168.137.31	TCP	1494	443 → 44652 [ACK] Se
6	0.057232	103.142.110.26	192.168.137.31	TCP	1494	[TCP Previous segmen
7	0.057296	103.142.110.26	192.168.137.31	TCP	1494	443 → 44652 [ACK] Se

Gambar 4. Pola Lalu Lintas Jaringan Public Group

2. *Private Group*

Private group merupakan grup yang hanya terlihat oleh orang yang sudah diundang dan hanya dapat bergabung melalui undangan (Inc., 2022). Percobaan pada grup privat ini sama seperti *channel* sebelumnya dengan mengirimkan pesan "TESTING CHAT" pada *private group* untuk mengetahui pola pesan jika dikirimkan melalui *channel private group* kemudian setiap pesan dicatat pola lalu lintas jaringannya menggunakan Wireshark.

Tercatat lalu lintas jaringan yang terlihat pada Gambar 5 memberikan informasi bahwa IP *client* (192.168.137.31) mengirimkan *Application Data* dengan paket sebesar 242 ke IP *server* (103.142.110.26) pada waktu ke 0 ketika target mengirim pesan pertama kali kemudian dilanjutkan pengiriman dari server ke *client* sebanyak dua kali dan juga tetap menggunakan protokol TLSv1.2. Pola berikutnya yang dapat terlihat yaitu pada waktu ke lima di mana dari *client* mengirim ke *server* dengan protokol TCP dengan paket sebesar 54 dengan proses ACK ke *port* 443 *server*.

	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.137.31	103.142.110.26	TLSv1.2	242	Application Data
2	0.039442	103.142.110.26	192.168.137.31	TLSv1.2	447	Application Data
3	0.039644	103.142.110.26	192.168.137.31	TLSv1.2	85	Application Data
4	0.041498	192.168.137.31	103.142.110.26	TLSv1.2	529	Application Data, Application Data,
5	0.082512	192.168.137.31	103.142.110.26	TCP	54	44652 → 443 [ACK] Seq=664 Ack=425 Wi
6	0.102736	103.142.110.26	192.168.137.31	TCP	54	443 → 44652 [ACK] Seq=425 Ack=664 Wi
7	0.104842	192.168.137.31	103.142.110.26	TLSv1.2	316	Application Data, Application Data

Gambar 5. Pola Lalu Lintas Jaringan Private Group

3. Personal Chat

Personal chat dilakukan dengan mengirimkan secara langsung pesan ke pada orang yang bersangkutan. Pola yang teridentifikasi pada pengiriman pesan menggunakan personal chat ini dapat dilihat pada gambar 6 yang menunjukan bahwa pola pengiriman *application data* dari IP client (192.168.137.31) ke IP *server* (103.142.110.26) dengan protokol TLSv1.2 dengan paket sebesar 112 pada waktu ke 0. Kemudian pada pengiriman dari IP *server* (103.142.110.26) ke IP *client* (192.168.137.31) pada nomor 3 menggunakan protokol TLSv1.2 untuk mengirim *application data* dengan mengirim paket sebesar 127. Pada nomor 5 melakukan pengiriman data dari IP *client* (192.168.137.31) ke IP *server* (103.142.110.26) menggunakan protokol TCP dengan paket sebesar 54.

	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.137.31	103.142.110.26	TLSv1.2	112	Application Data
2	0.021535	103.142.110.26	192.168.137.31	TLSv1.2	127	Application Data
3	0.022065	103.142.110.26	192.168.137.31	TLSv1.2	127	Application Data
4	0.038502	192.168.137.31	103.142.110.26	TLSv1.2	302	Application Data
5	0.040108	192.168.137.31	103.142.110.26	TCP	54	44628 → 443 [ACK]
6	0.040395	192.168.137.31	103.142.110.26	TCP	54	44632 → 443 [ACK]
7	0.078555	103.142.110.26	192.168.137.31	TLSv1.2	595	Application Data

Gambar 6. Pola Lalu Lintas Jaringan Personal Message

3. HASIL DAN PEMBAHASAN

Aplikasi Synology Chat ini menggunakan protokol TLSv1.2 (*Transport Layer Security)* yaitu protokol enkripsi yang digunakan untuk mengirim data di internet (Ali, 2021). Protokol ini mengenkripsi *payload*, jadi hanya besar paket yang terlihat dan dari informasi tersebut dapat dianalisis polanya. Pola yang terlihat berbeda antar *channel public group*, *private group* dan *personal chat.* Setiap *channel* memiliki pola yang berbeda yang ditunjukan pada besarnya Lenght, yaitu suatu nilai dari paket data termasuk header dengan besaran byte.

1. Pola pada *public group*

Pada *public group* dapat dilihat bahwa pola 2 dan 3 memiliki kesamaan pada alamat pengiriman IP *client* ke *server* maupun sebaliknya dengan protokol yang digunakan dan besar paket yang dikirimkan tersebut sama adapun perbedaan besar paket setelah mengirim paket sebesar 112 pada pola 2 dan 3, seperti ditunjukan pada gambar 7.

Public group							
Pola	No	Time	Source	Destination	Protocol	Leng	Info
1	1	0	192.168.137.31	103.142.110.26	TLSv1.2	185	Application Data
	2	0.056606	103.142.110.26	192.168.137.31	TCP	1494	443 → 44652 [ACK] Seq=1
	3	0.056717	103.142.110.26	192.168.137.31	TCP	1494	443 → 44652
	4	0.056777	103.142.110.26	192.168.137.31	TCP	1494	443 → 44652
	5	0.056823	103.142.110.26	192.168.137.31	TCP	1494	443 → 44652
2	422	24.440079	192.168.137.31	103.142.110.26	TLSv1.2	112	Application Data
	423	24.446979	192.168.137.31	103.142.110.26	TLSv1.2	242	Application Data
	424	24.460776	103.142.110.26	192.168.137.31	TLSv1.2	127	Application Data
	425	24.462544	192.168.137.31	103.142.110.26	TCP	54	44658 → 443
	426	24.489981	103.142.110.26	192.168.137.31	TLSv1.2	85	[TCP Previous
3	1004	58.853374	192.168.137.31	103.142.110.26	TLSv1.2	112	Application Data
	1005	58.857558	192.168.137.31	103.142.110.26	TLSv1.2	299	Application Data
	1006	58.87555	103.142.110.26	192.168.137.31	TLSv1.2	127	Application Data
	1007	58.878474	192.168.137.31	103.142.110.26	TCP	54	44658 → 443
	1008	58.909301	103.142.110.26	192.168.137.31	TLSv1.2	85	[TCP Previous

Gambar 7. Pola yang Teridentifikasi pada Public Group

2. Pola pada *private group*

Pola yang teridentifikasi pada *private group* sedikit berbeda daripada *channel* yang lain yaitu pada besar paket yang dikirim. Pada Pola 1 dan 3 memiliki besar paket yang sama yang dikirimkan dari IP *server* ke *client* yaitu sebesar 447 dan 85 kemudian pada pengiriman menggunakan TCP besar paket menjadi 54. Terlihat semua pola dominan menggunakan protokol TLSv1.2 dalam mengirimkan *application data* dari *client* ke *server* maupun sebaliknya, seperti terlihat pada gambar 8.

Private group							
Pola	No	Time	Source	Destination	Protocol	Lengt	Info
1	1	0	192.168.137.31	103.142.110.26	TLSv1.2	242	Application
	2	0.039442	103.142.110.26	192.168.137.31	TLSv1.2	447	Application Data
	3	0.039644	103.142.110.26	192.168.137.31	TLSv1.2	85	Application
	4	0.041498	192.168.137.31	103.142.110.26	TLSv1.2	529	Application
	5	0.082512	192.168.137.31	103.142.110.26	TCP	54	44652 → 443
2	439	40.145547	192.168.137.31	103.142.110.26	TLSv1.2	112	Application
	440	40.145547	192.168.137.31	103.142.110.26	TLSv1.2	210	Application
	441	40.168586	103.142.110.26	192.168.137.31	TLSv1.2	127	Application
	442	40.17006	192.168.137.31	103.142.110.26	TCP	54	44658 → 443
	443	40.183701	103.142.110.26	192.168.137.31	TCP	85	[TCP Previous
3	2095	77.485414	192.168.137.31	103.142.110.26	TLSv1.2	210	Application
	2096	77.525204	103.142.110.26	192.168.137.31	TLSv1.2	447	Application
	2097	77.525441	103.142.110.26	192.168.137.31	TLSv1.2	85	Application
	2098	77.547978	192.168.137.31	103.142.110.26	TLSv1.2	739	Application
	2099	77.584159	192.168.137.31	103.142.110.26	TCP	54	44652 → 443

Gambar 8. Pola yang Teridentifikasi pada Private Group

3. Pola pada *personal message*

Pada *personal chat* pola 1 dan 2 mempunyai pola yang sama dalam pengiriman *application data* dari IP *client* ke *server* maupun sebaliknya. Kemudian Lenght dari paket data yang dikirimkan sama yaitu pada pola 1 nomor 1 sebesar 112 dengan pola 2 nomor 352. Begitu juga dengan pola 1 nomor 3 dan pola 2 nomor 354 memiliki nilai Lenght dari paket yang sama yaitu 127. Protokol yang digunakan pun juga sama yaitu TLSv1.2. Perbedaan hanya pada pola ke 3 pada besaran Lenght dari paket yang dikirimkan dan pola IP setelahnya tetapi tetap juga menggunakan TLSv1.2 seperti terlihat pada gambar 9.

Personal Message							
Pola	No	Time	Source	Destination	Protocol	Leng	Info
1	1	0	192.168.137.31	103.142.110.26	TLSv1.2	112	Application Data
	2	0.021535	103.142.110.26	192.168.137.31	TLSv1.2	127	Application Data
	3	0.022065	103.142.110.26	192.168.137.31	TLSv1.2	127	Application Data
	4	0.038502	192.168.137.31	103.142.110.26	TLSv1.2	302	Application Data
	5	0.040108	192.168.137.31	103.142.110.26	TCP	54	44628 → 443
2	352	63.993192	192.168.137.31	103.142.110.26	TLSv1.2	112	Application Data
	353	64.009194	192.168.137.31	103.142.110.26	TLSv1.2	210	Application Data
	354	64.014484	103.142.110.26	192.168.137.31	TLSv1.2	127	Application Data
	355	64.016522	192.168.137.31	103.142.110.26	TCP	54	44628 → 443
	356	64.017799	103.142.110.26	192.168.137.31	TLSv1.2	127	Application Data
3	1244	95.451644	192.168.137.31	103.142.110.26	TLSv1.2	185	Application Data
	1245	95.495542	103.142.110.26	192.168.137.31	TCP	1494	443 → 44636
	1246	95.495722	103.142.110.26	192.168.137.31	TCP	1494	443 → 44636
	1247	95.495808	103.142.110.26	192.168.137.31	TCP	1494	443 → 44636
	1248	95.502711	192.168.137.31	103.142.110.26	TLSv1.2	586	Application Data,

Gambar 9. Pola yang Teridentifikasi pada Personal Message

4. KESIMPULAN DAN SARAN

Pesan instan Synology Chat telah memanfaatkan komunikasi pada jaringan yang terenkripsi berdasarkan pantauan menggunakan Wireshark dengan metode MITM. Oleh sebab itu, pemantauan yang dilakukan dengan mengidentifikasi IP *client*, *server*, besaran Lenght dari paket yang dikirimkan, *ports* dan *timeline* bahwa protokol yang digunakan yaitu TLSv1.2, yaitu data yang dikirimkan terenkripsi. Begitu pula dengan port yang digunakan juga 443 atau HTTPS pada komunikasi berbasis web. Pola yang ditemukan pada besar paket yang dikirimkan ke server maupun ke client beberapa dapat teridentifikasi polanya. Length pada *public group* dan *personal chat*besar paket hampir sama tetapi pada *private group* pola length pada data paket yang dikirimkan lebih besar daripada *channel* lain dan pengiriman dominan menggunakan protokol TLSv1.2. Dari penelitian ini didapatkan manfaat terutama untuk mengivestigasi kasus kriminal tetapi terbatas pada penemuan pola besar paket yang dikirimkan.

Saran yang dapat diberikan pada penelitian selanjutnya yaitu mengidentifikasi pola lain seperti ketika mengirimkan gambar, menelfon, *video call*, mengetik, saling berbalas pesan maupun saat membuka aplikasi. Proses ini dapat disebut dengan analisis forensik jaringan yang bermanfaat ketika melakukan investigasi tindak kejahatan. Penelitian ini juga terbatas pada versi aplikasi yang digunakan sehingga kedepannya hasil yang didapatkan bisa saja berbeda dan juga *smartphone* yang masih mempunyai *background* proses sehingga mempengaruhi pengamatan.

DAFTAR PUSTAKA

SIDIK, & PUTRA, M. (2018). Implementasi Network Attached Storage (NAS) Menggunakan Synology Disk Station Manager (DSM 5.2) Untuk Optimalisasi Data Sharing Center. Jurnal Teknik Komputer, IV(2), 39-47. doi:10.31294/jtk.v4i2.3508

POLWALNYCKY, D., BAGGILI, I., MARRINGTON, A., MOORE, J., & BREITINGER, F. (2015). Network and device forensic analysis of Android social-messaging applications. *Digital Investigation*, *14*, S77–S84. https://doi.org/10.1016/j.diin.2015.05.009

INC., S. (2022). *Creating Channels and Conversations | Synology Chat - Synology Knowledge Center*. Kb.synology.com. Retrieved 30 August 2022, fromhttps://kb.synology.com/enaf/DSM/help/Chat/chat_creating_channels_conversations?version=7.

ALI, I. (2021). *Examining cyber security implementation through TLS/SSL on academic institutional repository in Indonesia Irhamni Ali*. *17*(2), 238–249.

ANGLANO, C., CANONICO, M., & GUAZZONE, M. (2020). *Computers & Security The Android Forensics Automator (AnForA): A tool for the Automated Forensic Analysis of Android Applications*. *88*. https://doi.org/10.1016/j.cose.2019.101650

KARPISEK, F., BAGGILI, I., & BREITINGER, F. (2015). WhatsApp network forensics : Decrypting and understanding the WhatsApp call signaling messages. *Digital Investigation*, *15*, 110–118. https://doi.org/10.1016/j.diin.2015.09.002

SUDOZAI, M. A. K., SALEEM, S., BUCHANAN,

W. J., HABIB, N., & ZIA, H. (2018). Forensics study of IMO call and chat app. *Digital Investigation*, *25*, 5–23. https://doi.org/10.1016/j.diin.2018.04.006

YUSOFF, M.N YUSOFF, M.N.; DEHGHANTANHA, A.; MAHMOD, R. Network Traffic Forensics on Firefox Mobile OS: Facebook, Twitter, and Telegram as Case Studies. In Contemporary Digital Forensic Investigations of Cloudand Mobile Applications; Elsevier Inc.: Amsterdam, The Netherlands, 2017; pp. 63–78.

AFZAL, A., HUSSAIN, M., SALEEM, S., SHAHZAD, M. K., HO, A. T. S., & JUNG, K. (2021). *applied sciences Encrypted Network Traffic Analysis of Secure Instant Messaging Application : A Case Study of Signal Messenger App*.

CONTI, M., MANCINI, L., SPOLAOR, R., & VERDE, N. (2015). Can't You Hear Me Knocking. *Proceedings Of The 5Th ACM Conference On Data And Application Security And Privacy*. https://doi.org/10.1145/2699026.2699119

RATHI, K.; KARABIYIK, U.; ADERIBIGBE, T.; CHI, H. (2018). Forensic analysis of encrypted instant messaging applications on Android. International Symposium on Digital Forensic and Security (ISDFS), Antalya, Turkey, 22–25March 2018.

MALLIK, A. (2018). *MAN-IN-THE-MIDDLE-ATTACK : UNDERSTANDING IN SIMPLE.*, Jurnal Pendidikan Teknologi Informasi 2, 2,109–134.

HE, G.; XU, B.; ZHU, H (2017). Identifying Mobile Applications for Encrypted Network Traffic. In Proceedings of the 2017 Fifth International Conference on Advanced Cloud and Big Data (CBD), Shanghai, China, 13–16 August 2017; pp. 279–284.

OMOLARA, A.E.; JANTAN, A.; ABIODUN, O.I.; DADA, K.V.; ARSHAD, H.; EMMANUEL, E. A Deception Model Robust to Eavesdropping Over Communication for Social Network Systems 2019.IEEE. Retrieved 5 September 2022, from https://ieeexplore.ieee.org/document/8760227.

AWAN, F.A. Forensic examination of social networking applications on smartphones (2015). In Proceedings of the 2015 Conference on Information Assurance and Cyber Security (CIACS), Rawalpindi, Pakistan, 18 December 2015; pp. 36–43.

GREGORIO, J., GARDEL, A., & ALARCOS, B. (2017). Forensic analysis of Telegram Messenger for Windows Phone. *Digital Investigation*, *22*, 88–106. https://doi.org/10.1016/j.diin.2017.07.004