Journal of System & Technology
e-ISSN: XXXX-XXXX
Vol. 1, No. 1, Juni 2025

ANALISIS KEAMANAN PROTOKOL TELNET VS SSH MENGGUNAKAN WIRESHARK DALAM JARINGAN TERBUKA

Achmad Haikal Fikri[1]*, Adinda Aulia Rahmi[1], Rakhmadi Rahman[1]
[1]Institut Teknologi Bacharuddin Jusuf Habibie, Parepare, Sulawesi Selatan, Indonesia
achmadhaikal1405@gmail.com, adindaauliar11@gmail.com, rakhmadi.rahman@ith.ac.id
*Corresponding author: achmadhaikal1405@gmail.com

Abstract: Keamanan jaringan merupakan aspek yang krusial dalam teknologi informasi, khususnya dalam komunikasi jarak jauh seperti remote login. Remote login merupakan metode yang memungkinkan pengguna untuk mengakses sistem komputer lain dari jarak jauh. Penelitian ini menganalisis secara komparatif keamanan dua protokol, Telnet dan SSH, dalam jaringan terbuka. Metode yang digunakan pada penelitian ini adalah metode eksperimen dengan pendekatan prototyping. Simulasi dilakukan dengan memanfaatkan Wireshark dalam WSL Kali Linux. Hasil menunjukkan Telnet mengirim data dalam bentuk plaintext, rentan terhadap penyadapan. Sebaliknya, SSH mengenkripsi data, membuatnya lebih aman. Oleh karena itu, SSH lebih direkomendasikan untuk komunikasi jarak jauh yang aman.

Keywords—Enkripsi, Keamanan Jaringan, Remote Login, SSH, Telnet

I. PENDAHULUAN

Teknologi informasi dan komunikasi saat ini telah berkembang sangat pesat, sehingga memberikan dampak yang signifikan terhadap sistem jaringan, khususnya dalam komunikasi jarak jauh seperti remote login. Remote login memungkinkan pengguna untuk mengakses dan mengontrol sistem komputer dari lokasi yang berbeda secara real-time. Namun, praktik ini juga menimbulkan tantangan serius terhadap keamanan jaringan [1], [2].

Salah satu protokol yang digunakan untuk remote login adalah Telnet, yang bekerja pada port 23 dan memungkinkan pengguna untuk berkomunikasi dengan sistem jarak jauh melalui antarmuka perintah. Kelemahan utama dari Telnet adalah tidak digunakannya enkripsi dalam proses komunikasi [3, 4]. Hal ini membuat informasi seperti username dan password mudah diakses oleh pihak tidak berwenang. Akibatnya, Telnet sangat rentan terhadap penyadapan (*sniffing*), serangan *man-in-the-middle* (MitM), dan berbagai bentuk eksploitasi lainnya [5], [6].

Sebagai solusi atas kelemahan Telnet, dikembangkan protokol Secure Shell (SSH). SSH merupakan protokol jaringan yang lebih aman karena menyediakan mekanisme enkripsi dan autentikasi dua arah yang kuat [7]. SSH beroperasi pada port 22 dan mendukung autentikasi menggunakan kata sandi yang kuat maupun kunci publik, serta menyediakan komunikasi data terenkripsi antara dua komputer yang terhubung melalui jaringan terbuka [2], [8]. SSH merupakan protokol kriptografi yang terdiri dari rangkaian utilitas yang mengimplementasikan standar keamanan tinggi [4].

Beberapa studi terdahulu telah menunjukkan bahwa SSH lebih unggul dibandingkan Telnet dalam menjaga kerahasiaan dan integritas data. Protokol Telnet tidak menggunakan enkripsi, sehingga paket data yang dikirim dapat dengan mudah dilihat dalam bentuk teks biasa [3], [4]. Untuk akses ke perangkat jaringan seperti router, penggunaan SSH lebih disarankan karena mendukung enkripsi [9], [10]. Hal ini menegaskan kembali bahwa Telnet yang tidak menggunakan enkripsi dapat dengan mudah dimanfaatkan untuk mencuri informasi yang dikirimkan melalui jaringan [11].

Oleh karena itu, sangat penting dilakukan pengujian langsung untuk membandingkan kedua protokol, yaitu Telnet dan SSH, dalam konteks jaringan terbuka. Tujuan penelitian ini adalah untuk menganalisis secara empiris keamanan kedua protokol tersebut melalui simulasi koneksi remote login yang dipantau menggunakan perangkat sniffing Wireshark. Metode ini diharapkan dapat memberikan bukti konkret tentang kelemahan Telnet dan keunggulan SSH dalam melindungi data dari ancaman penyadapan di jaringan yang tidak aman [12], [13].

II. METODOLOGI

Penelitian ini menggunakan pendekatan eksperimen dengan model prototyping. Metode ini dipilih karena memungkinkan pengamatan langsung terhadap hasil pengujian dua protokol komunikasi jarak jauh, yaitu Telnet dan SSH, dalam konteks jaringan terbuka. Simulasi jaringan dibangun menggunakan Windows Subsystem for Linux (WSL) dengan distribusi Kali Linux, yang digunakan untuk menjalankan alat sniffing dan analisis data menggunakan Wireshark. Skenario ini merepresentasikan kondisi jaringan terbuka di mana komunikasi data dapat disadap oleh pihak ketiga menggunakan teknik sniffing. Tools yang digunakan antara lain:

1) KpyM Telnet Server v1.19

Gambar 1. Kpym Telnet Server

Perangkat lunak ini (gambar 1) memungkinkan pengaktifan layanan Telnet pada sistem operasi Windows. Digunakan untuk melakukan simulasi koneksi Telnet dari klien ke server.

2) OpenSSH Server v9.x di Windows

Gambar 2. OpenSSH

OpenSSH (gambar 2) merupakan implementasi protokol SSH berbasis open-source yang diaktifkan melalui PowerShell pada Windows. Digunakan untuk mengaktifkan layanan SSH untuk pengujian koneksi aman.

3) Wireshark v4.x dan WSL Kali Linux (WSL 2023.4)

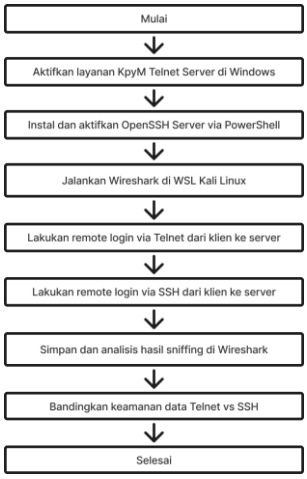
Gambar 3. Wireshark & WSL Kali Linux

Wireshark (gambar 3) digunakan sebagai alat sniffing untuk menangkap dan menganalisis paket data. Kali Linux berfungsi sebagai lingkungan pengujian yang dilengkapi dengan tool keamanan jaringan.

Langkah-langkah pengujian penelitian yang dilakukan adalah sebagai berikut:

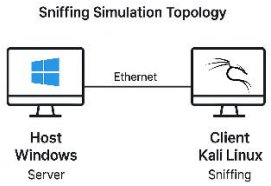
1) Mengaktifkan layanan KpyM Telnet Server di Windows.
2) Menginstal dan mengaktifkan OpenSSH Server di Windows melalui PowerShell.
3) Menginstal dan menjalankan Wireshark di dalam WSL Kali Linux.
4) Melakukan koneksi remote login menggunakan protokol Telnet dari klien ke server.
5) Melakukan koneksi remote login menggunakan protokol SSH dari klien ke server.
6) Melakukan sniffing menggunakan Wireshark selama proses koneksi berlangsung.
7) Menganalisis hasil sniffing di Wireshark, khususnya pada isi payload paket untuk memastikan apakah data sensitif seperti username dan password dapat terbaca.
8) Membandingkan hasil analisis antara protokol Telnet dan SSH dari sisi keamanan data yang dikirimkan di jaringan.

Untuk memperjelas alur pengujian, berikut disajikan diagram alir atau *flowchart* (gambar 4) dari proses simulasi dan analisis protokol Telnet dan SSH:



Gambar 4. Flowchart Prosedur Pengujian Keamanan Protokol Telnet dan SSH

Adapun topologi jaringan pengujian digambarkan sebagai berikut:



Gambar 5. Topologi Simulasi Sniffing

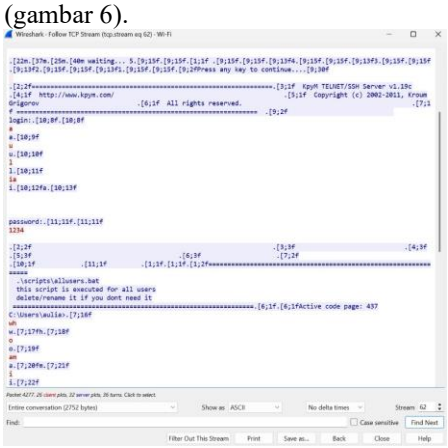
Topologi simulasi (gambar 5) terdiri dari satu host Windows sebagai server dan satu klien berbasis Kali Linux, di mana proses sniffing dilakukan dari sisi klien.

Etika: Seluruh pengujian dilakukan di lingkungan simulasi lokal tanpa melibatkan data pengguna aktual, sehingga tidak menimbulkan risiko keamanan atau privasi.

III. HASIL DAN PEMBAHASAN

A. Hasil dan Pembahasan Pengujian Koneksi Telnet

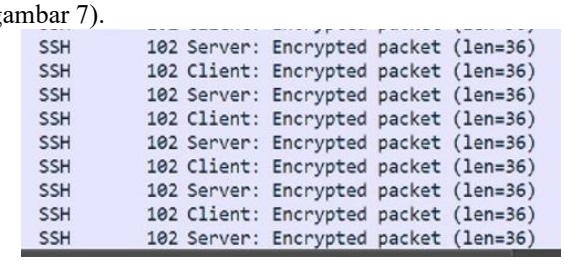
Pengujian koneksi Telnet menunjukkan bahwa seluruh data yang ditransmisikan, termasuk username, password, dan perintah terminal, ditampilkan dalam bentuk teks biasa (plaintext) pada hasil tangkapan paket di Wireshark. Hal ini membuktikan bahwa protokol Telnet tidak menggunakan enkripsi dalam proses transmisinya, sehingga sangat rentan terhadap serangan penyadapan atau sniffing (gambar 6).



Gambar 6. Hasil pengujian koneksi Telnet

B. Hasil dan Pembahasan Pengujian Koneksi SSH

Pada sisi lain, pengujian koneksi SSH menghasilkan data yang seluruhnya terenkripsi. Proses handshake TCP berhasil dilakukan, dan setelahnya semua paket ditandai sebagai protokol SSH dengan isi payload dalam bentuk heksadesimal yang tidak terbaca. Tidak ditemukan informasi plaintext seperti username maupun password, baik pada awal koneksi maupun selama sesi berlangsung (gambar 7).



Gambar 7. Hasil Tangkapan Wireshark pada Protokol SSH

Tangkapan layar menunjukkan bahwa seluruh paket komunikasi dalam sesi SSH ditandai sebagai “Encrypted packet”, baik dari sisi server maupun klien.

C. Perbandingan Protokol Telnet dan SSH

Berdasarkan hasil pengujian terhadap kedua protokol, diperoleh perbandingan sebagaimana pada tabel 1:

Tabel 1. Perbandingan Protokol Telnet dan SSH

Protokol	Payload Terbaca	Enkripsi	Keterangan
Telnet	Ya	Tidak	Data username, password, dan command dapat dibaca langsung di Wireshark
SSH	Tidak	Ya	Semua data terenkripsi, hanya terlihat sebagai paket acak di Wireshark

SSH menggunakan mekanisme enkripsi end-to-end. Protokol ini biasanya mengimplementasikan algoritma Diffie-Hellman untuk pertukaran kunci, RSA atau ECDSA untuk autentikasi, dan Advanced Encryption Standard (AES) untuk enkripsi simetris. Selain itu, Hash-based Message Authentication Code (HMAC) digunakan untuk menjamin integritas data. Berbeda dengan SSH, Telnet tidak menggunakan mekanisme enkripsi apa pun. Semua data dikirim dalam bentuk terbuka, yang dapat dengan mudah diakses oleh penyerang dalam jaringan.

D. Implikasi Keamanan

Dari hasil ini, dapat disimpulkan bahwa Telnet membuka peluang besar terhadap serangan seperti sniffing, session hijacking, dan man-in-the-middle (MitM). SSH secara substansial menutup celah tersebut melalui enkripsi yang kuat dan verifikasi identitas dua arah. Hasil pengujian ini mendukung temuan studi sebelumnya yang menyarankan penghapusan Telnet dari sistem produksi dan penggunaan SSH sebagai standar minimum keamanan jaringan.

IV. Implikasi Penelitian

Penelitian ini memiliki implikasi praktis dan akademik:

1) Praktis: Penggunaan Telnet sebaiknya dihentikan pada jaringan terbuka. SSH harus menjadi standar minimum untuk komunikasi jarak jauh.
2) Akademik: Penelitian ini mendukung pentingnya integrasi praktik keamanan jaringan nyata seperti sniffing ke dalam kurikulum pendidikan, serta membuka peluang studi lanjut terhadap protokol lainnya dan konfigurasi keamanan tingkat lanjut.

V. Keterbatasan Penelitian

Penelitian ini memiliki dua implikasi utama. Secara praktis, hasil menunjukkan bahwa Telnet tidak layak digunakan di jaringan terbuka karena rawan penyadapan, sehingga SSH perlu dijadikan standar minimum komunikasi jarak jauh yang aman [3], [4], [8]. Secara akademik, penelitian ini menegaskan pentingnya integrasi praktik analisis keamanan jaringan berbasis sniffing seperti Wireshark dalam proses pembelajaran [12], [13], serta membuka peluang studi lanjut terhadap protokol lain dan konfigurasi keamanan lanjutan seperti SSH key-based authentication [2],[11].

VI. Rekomendasi Penelitian Selanjutnya

Penelitian selanjutnya disarankan untuk memperluas ruang lingkup pengujian dengan beberapa arah sebagai berikut:

1) Melakukan pengujian pada jaringan publik atau berbasis cloud untuk melihat efektivitas protokol dalam skenario dunia nyata.
2) Membandingkan protokol komunikasi lain seperti FTP vs SFTP, atau RDP vs SSH.
3) Menguji konfigurasi lanjutan SSH, termasuk key-based authentication, port forwarding, dan multi-factor authentication.
4) Mengintegrasikan SSH dengan sistem keamanan lainnya seperti IDS, firewall, atau SIEM untuk evaluasi pertahanan berlapis.
5) Menggunakan tools tambahan seperti Ettercap, Nmap, atau TCPdump untuk eksplorasi jenis serangan lain.
6) Melakukan evaluasi performa dari sisi efisiensi penggunaan sumber daya, latensi, dan bandwidth.

VII. Kesimpulan

Berdasarkan hasil pengujian dan analisis terhadap protokol Telnet dan SSH dalam jaringan terbuka, dapat disimpulkan bahwa Telnet tidak menyediakan mekanisme enkripsi, sehingga seluruh data yang ditransmisikan sangat rentan terhadap penyadapan. Sebaliknya, SSH menyediakan enkripsi end-to-end yang efektif melindungi data selama proses komunikasi. Kontribusi utama dari penelitian ini adalah demonstrasi praktis perbedaan tingkat keamanan kedua protokol dengan menggunakan Wireshark dalam lingkungan simulasi yang mendekati skenario nyata. Hasil ini memperkuat pentingnya penggunaan protokol aman seperti SSH dalam jaringan modern, terutama untuk aktivitas remote login.

VIII. Saran

Untuk komunikasi jarak jauh yang melibatkan data sensitif, penggunaan protokol SSH sangat disarankan karena menyediakan tingkat keamanan yang lebih tinggi dibandingkan Telnet. Penggunaan Telnet sebaiknya dibatasi hanya pada jaringan internal yang tertutup dan telah dilindungi oleh firewall serta sistem keamanan lainnya. Selain itu, administrator jaringan perlu diberikan edukasi mengenai risiko penggunaan protokol yang tidak aman. Tools seperti Wireshark dapat dijadikan media pembelajaran dalam kurikulum keamanan jaringan untuk memberikan pemahaman nyata tentang pentingnya enkripsi dalam komunikasi data.

Daftar Pustaka

[1] J. Park, J. Kim, B. B. Gupta and N. Park, "Network Log-Based SSH Brute-Force Attack Detection Model," Computers, Materials and Continua, vol. 68, no. 1, pp. 887-901, 2021.
[2] A. Upadhyay, S. Kulkarni, S. Roy, K. Yuvaraj, V. Ashok and S. D. Kale, "Understanding the Limitations of Secure Shell (SSH) in Wireless Network Security," International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON), pp. 1-7, 2023.

[3] M. L. A. Pérez and I. A. Muñoz, "Practical evaluation of the Telnet and SSH protocols," Gestión Competitividad E Innovación, vol. 9, no. 1, pp. 17-29, 2021.
[4] M. Başer, E. Y. Güven and M. A. Aydın, "SSH and Telnet Protocols Attack Analysis Using Honeypot Technique: Analysis of SSH and TELNET Honeypot," IEEE, 2021.
[5] G. Oh, "Vulnerability Analysis and Demonstration of FTP and Telnet Access in Drone Environment: Based on Product A," KoreaScience, pp. 197-198, 2023.
[6] L. Huang, " Research and Implementation of Computer Laboratory Network Management Tool Based on Telnet Protocol," Journal of Physics: Conference Series, vol. 1744, 2021.
[7] A. M. Shabalin and E. A. Kaliberda, "Development of a Set of Procedures for Providing Remote Access to a Corporate Computer Network by means of the SSH Protocol (Using the Example of the CISCO IOS Operating System)," IEEE, 2021.
[8] S. Sentanoe and H. P. Reiser, "SSHkex: Leveraging virtual machine introspection for extracting SSH keys and decrypting SSH network traffic," Forensic Science International: Digital Investigation, vol. 40, 2022.
[9] P. Plesowicz, "A15: Secure signal tunneling for SCADA and PLCs using SSH protocol," IFAC Proceedings Volumes, vol. 37, no. 20, pp. 88-93, 2024.
[10] F. Baumer, M. Brinkmann and J. Schwenk, "Terrapin Attack: Breaking SSH Channel Integrity By Sequence Number Manipulation," Cryptography and Security (cs.CR), vol. 2, pp. 1-21, 2023.
[11] F. Michel and O. Bonaventure, "Towards SSH3: how HTTP/3 improves secure shells," Networking and Internet Architecture (cs.NI), pp. 1-7, 2023.
[12] S. Toledo, "SSH tunneling to connect to remote computers," Software Impacts, vol. 17, 2023.
[13] A. S. Syahab, E. I. H. Ujianto and Rianto, "Penggunaan Wireshark dan NESSUS untuk Analisis SSL/TLS pada Keamanan Data Pengguna Website," Jurnal Informatika, vol. 7, no. 2, pp. 183-192, 2023.
[14] W. Agustiara, A. Pratama and S. Junaidi, "Analisis Keamanan Protokol Secure Socket Layer Terhadap Serangan Packet Sniffing Pada Website Portal Berita Harian Umum Korban Padang," Jurnal Teknik Informatika Kaputama (JTIK), vol. 6, no. 1, pp. 10-15, 2022.
[15] "Integration of Low Interaction Honeypot and ELK Stack as Attack Detection Systems on Servers," Jurnal Penelitian Pos dan Informatika, vol. 11, no. 1, pp. 19-29, 2021.
[16] D. F. Arkansyah, "Studi Kasus Antisipasi Ancaman Keamanan Internet terhadap Layanan Internet Satelit di PT Bis Data Indonesia untuk Proyek BAKTI menggunakan Layer-7 Deep Packet Inspection," Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer, vol. 7, no. 3, pp. 1353-1365, 2023.
[17] L. Z. a. F. X. Fu, "Lightweight SSH protocol enhancement for IoT devices in hostile environments," Journal of Network and Computer Applications, 2023.
[18] B. Choi, "Cisco IOS, Linux, TFTP, and Telnet Labs. In: Introduction to Python Network Automation," Springer Nature Link, vol. 1, pp. 753-802, 2024.
[19] D. A. Rachman, Y. Muhyidin and M. A. Sunandar, "Analisis Kualitas Layanan Jaringan Internet Fiber to the Home PT. XYZ Menggunakan Wirechark," Jurnal Ilmiah Teknik dan Ilmu Komputer, vol. 2, no. 3, pp. 214-222, 2023.