Benchmarking Post-quantum Public Key Encryption in TLS1.3 A Case Study on Kyber

Jingzhe Wang

Open Source:

- Data Source
- Project File

Figure

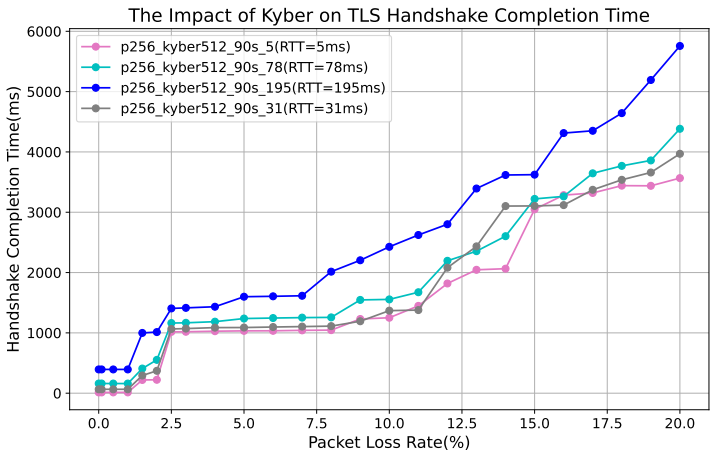

Figure 1: TLS Handshake Completion Time v.s. Packet Loss Rate

Legend Explained:

- The pink line reports Kyber's performance where we set RTT(Round-trip Time) = $5ms$.
- The light blue line reports Kyber's performance where we set RTT = $78ms$.
- The deep blue line reports Kyber's performance where we set RTT = $195ms$.
- The gray line reports Kyber's performance where we set RTT = $31ms$.

Findings:

- This is a multi-line plot that investigates the impact of Kyber on TLS handshake completion time.
- This plot reflects the performance of Kyber under four different RTT settings.
- By adopting Kyber as the core public key construction in TLS, with the increasing packet loss rate, the handshake completion time becomes longer. This observation works for all four settings.

Figure

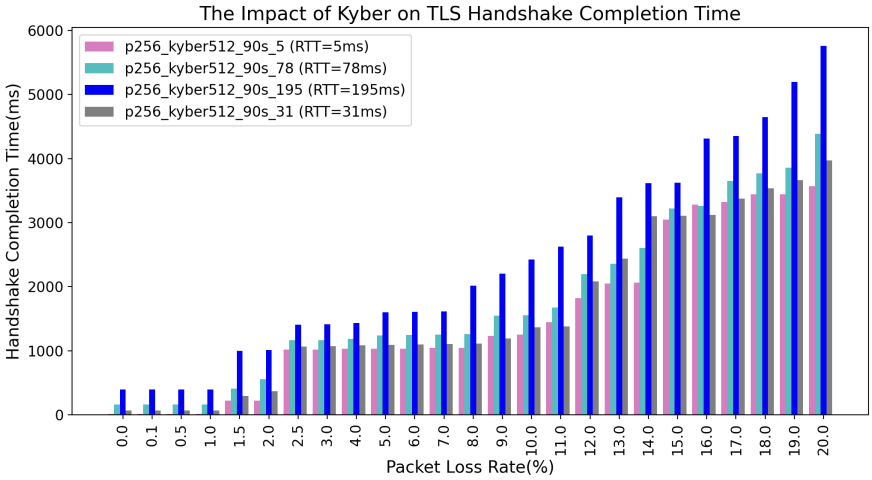

Figure 2: TLS Handshake Completion Time v.s. Packet Loss Rate (**Bar Chart**)

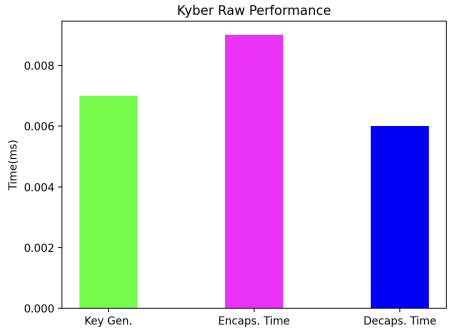

Figure 3: Kyber Raw Performance Result (Bar Chart)

Legend Explained: Under each packet loss setting, in **Figure 2**, we have the following: (1) The pink bar reflects Kyber's performance where we set RTT = $5ms$; (2) The light blue bar reports Kyber's performance where we set RTT = $78ms$; (3) The deep blue line captures Kyber's performance where we set RTT = $195ms$; (4) The gray line reports Kyber's performance where we set RTT = $31ms$. To reflect the raw performance of Kyber, in **Figure 3**, the green bar reports the key generation time of Kyber, the magenta bar gives us the encapsulation time of Kyber, and the blue bar outputs the decapsulation time of Kyber.

Finding Texts:(1) **Figure 1** aims at comparing the four Kyber settings under each packet loss rate. Compared with the line chart, this one says clearly the impact of RTT when we fix the packet loss rate. Specifically, in almost each packet loss setting, higher RTT results in longer handshake completion time; when we observe the figure horizontally, we can see that with the increment of packet loss rate, when fixing RTT, the handshake completion time of Kyber grows; (2) **Figure 2** visualizes the raw performance of Kyber, including key generation time, encapsulation time, and decapsulation time. It is easy to see that the encapsulation time of Kyber dominates the raw performance metrics.

Aesthetic Considerations: In **Figure 2**, to make the x-ticks more clear, I enable the option that rotates the ticks vertically .

Figure

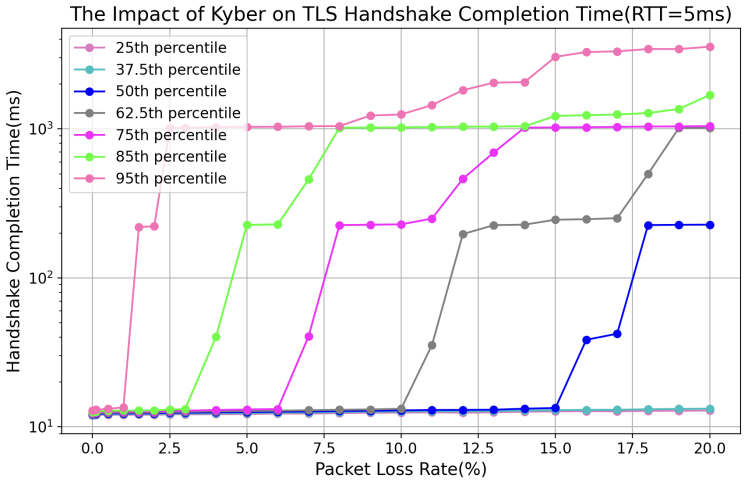

Figure 4: TLS Handshake Completion Time v.s. Packet Loss Rate (Various Percentile) - 5 RTT

Legend Explained: Figure 4 was visualized under $RTT = 5ms$.(1) The hot pink bar reflects 25th percentile value in terms of completion time; (2) The light blue bar reports 37.5th percentile value of Kyber's completion time; (3) The deep blue line captures Kyber's 50th percentile value;(4) The gray line reports Kyber's 62.5th percentile value; (5) The magenta line reports Kyber's 75th percentile value; (6) The lime line reports Kyber's 85th percentile value; (7) The deep pink bar reports Kyber's 95th percentile value

Findings: (1) This figure is used to visualize the distribution of the measured completion time under different packet loss settings. It is represented in a log scale for better visualization; (2) Observing this figure horizontally, we can see that a higher packet loss rate incurs significant variations in terms of handshake completion time, as shown in the case where the packet loss rate is 20%.

Data and method text describing the data and method used in this process: Technically, matplotlib.plot is the main environment that creates all the above figures. Specifically, plt.bar() governs the bar plots , and plt.plot() yields the line figures. In addition, plt.grid is enabled for the line figures. To make visualization clearer, I carefully picked four unique colors and enabled the log scale option. All data is automatically loaded by adopting pandas's read csv approach.

Significance statement on why the presented figure is important: Those figures pick one algorithm candidate and perform fine-grained visualization on the critical performance metrics, including raw performance and hand-shake time. The results can guide developers on whether Kyber is a good option for their application environment.

Improvements over the midterm version:(1) **Data Intensity**: Improvements regarding this point are two-fold: first, compared to my midterm version, in the bar chart visualization, I added new data reflecting raw performance results. Though it is not large scale, it shows critical performance metrics; second, in the midterm version, I only choose 95th percentile data to show the results; while in the percentile data visualization, I visualize all relevant data of Kyber. (2)**Creativity:** The bar plots reflect this point. Specifically, I adopt the skills in our class to alternatively visualize the data used in midterm in terms of the bar charts. This new visualization showcases more fine-grained results, such as reflecting the algorithms performances under different RTTS when fixing packet loss rate.(3)**Aesthetic Designs**: Related improvements on this point are as follows: first, in **Figure 2**, I vertically rotate the xticks to make the visualization readable; second, in the percentile figure, instead of using the default y-scale settings, I enable the log-scale option, showing the data clear.