

```

1  """
2  FTPChat - Encrypted FTP-based Messaging Protocol
3  Version 1.4 - November 2025
4  Type: Custom-styled MIT LICENSE
5  Author: Ahmed Omar Saad
6  Contact: ahmedomardev@outlook.com
7  FTPChat © 2025/5/20 by Ahmed Omar Saad is licensed under CC BY-NC-SA 4.0. To view a
  copy of this license, visit https://creativecommons.org/licenses/by-nc-sa/4.0/
8
9  *Permission is hereby granted, free of charge, to any person obtaining a copy
10 *of this software and associated documentation files (the "Software"), to deal
11 *in the Software without restriction, including without limitation the rights
12 *to use, copy, modify, merge, publish, distribute, and/or sublicense copies of
13 *the Software, subject to the following conditions:
14
15 *- The above copyright notice and this permission notice shall be included in all copies
  or substantial portions of the Software.
16 *- **Commercial use of this software requires prior written permission from the author.**
17 *- **The author reserves the right to relicense this software as closed-source or
  commercial at any time.**
18 *- All rights to the name "FTPChat" and its project specification are retained by Ahmed
  Omar Saad.
19
20 *THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
21 *INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
  PARTICULAR PURPOSE
22 *AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR
  ANY CLAIM,
23 *DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
  ARISING FROM,
24 *OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
25
26 *Notes:
27 *The author may offer separate commercial licenses for enterprise or closed-source use.
  Contact ahmedomardev@outlook.com for inquiries.
28 *This license applies to all source code, documentation, and project specifications
  included in the FTPChat project.
29 """
30
31 from datetime import datetime
32 from ftplib import FTP, FTP_TLS
33 from threading import Thread
34 from webbrowser import open as open_link
35
36 from ttkbootstrap import (
37     Window,
38     Button,
39     Label,
40     Entry,
41     Menu,
42     NORMAL,
43     DISABLED,
44     StringVar,
45     ScrolledText,
46 )
47 from tkinter import messagebox
48 from zlib import compress, decompress
49 from base64 import b64encode, b64decode
50
51
52 def help_func():
53     open_link("ahmed-omar-software-projects.mydurable.com")
54
55
56 # !The start of the characters list and key for mac encryption layers
57
58

```

```
59  CHARACTERS = (
60      "a" /
61      "b" /
62      "c" /
63      "d" /
64      "e" /
65      "f" /
66      "g" /
67      "h" /
68      "i" /
69      "j" /
70      "k" /
71      "l" /
72      "m" /
73      "n" /
74      "o" /
75      "p" /
76      "q" /
77      "r" /
78      "s" /
79      "t" /
80      "u" /
81      "v" /
82      "w" /
83      "x" /
84      "y" /
85      "z" /
86      "A" /
87      "B" /
88      "C" /
89      "D" /
90      "E" /
91      "F" /
92      "G" /
93      "H" /
94      "I" /
95      "J" /
96      "K" /
97      "L" /
98      "M" /
99      "N" /
100     "O" /
101     "P" /
102     "Q" /
103     "R" /
104     "S" /
105     "T" /
106     "U" /
107     "V" /
108     "W" /
109     "X" /
110     "Y" /
111     "Z" /
112     "!" /
113     "'" /
114     "#" /
115     "$" /
116     "%" /
117     "&" /
118     " " /
119     "(" /
120     ")" /
121     "*" /
122     "+" /
123     "/" /
124     "-" /
125     "." /
```

```
126    "/" ,
127    ":" ,
128    ";" ,
129    "<" ,
130    "=" ,
131    ">" ,
132    "?" ,
133    "@" ,
134    "[" ,
135    "\\\" ,
136    "]" ,
137    "^" ,
138    "-" ,
139    "`" ,
140    "{" ,
141    "|" ,
142    "}" ,
143    "~" ,
144    "0" ,
145    "1" ,
146    "2" ,
147    "3" ,
148    "4" ,
149    "5" ,
150    "6" ,
151    "7" ,
152    "8" ,
153    "9" ,
154 )
155 KEY_FOR_MAC_1 = (
156    "!" ,
157    ")" ,
158    "." ,
159    "3" ,
160    "(" ,
161    "J" ,
162    "b" ,
163    "C" ,
164    "A" ,
165    ";" ,
166    "n" ,
167    "_" ,
168    "$" ,
169    "0" ,
170    "," ,
171    "Z" ,
172    "B" ,
173    "c" ,
174    "|" ,
175    "g" ,
176    "7" ,
177    "^" ,
178    "'" ,
179    "I" ,
180    "v" ,
181    "P" ,
182    "Y" ,
183    "M" ,
184    "i" ,
185    "O" ,
186    "k" ,
187    "L" ,
188    "%" ,
189    "Y" ,
190    "D" ,
191    "#" ,
192    "@" ,
```

```
193     "<" /
194     "9" /
195     "r" /
196     "K" /
197     "2" /
198     "-" /
199     "t" /
200     "8" /
201     "H" /
202     "=" /
203     "+" /
204     "`" /
205     "o" /
206     "T" /
207     "u" /
208     "s" /
209     "p" /
210     "U" /
211     "S" /
212     "Q" /
213     "~" /
214     "!" /
215     "G" /
216     "*" /
217     "&" /
218     ">" /
219     "X" /
220     "a" /
221     "N" /
222     "4" /
223     "d" /
224     "m" /
225     "l" /
226     "v" /
227     "w" /
228     "z" /
229     ":" /
230     "q" /
231     "?" /
232     "5" /
233     "x" /
234     "{" /
235     "/" /
236     "F" /
237     "j" /
238     "]" /
239     "6" /
240     "W" /
241     "f" /
242     "R" /
243     "1" /
244     "}" /
245     "e" /
246     "[" /
247     "\\\" /
248     "h" /
249     "E" /
250 )
251 KEY_FOR_MAC_2 = (
252     "$" /
253     "i" /
254     "o" /
255     "c" /
256     "&" /
257     "L" /
258     "/" /
259     "~" /
```

260 "O" /  
261 "@" /  
262 "(" /  
263 "[" /  
264 "Q" /  
265 "v" /  
266 "a" /  
267 "Y" /  
268 "7" /  
269 "3" /  
270 "\*" /  
271 "X" /  
272 "x" /  
273 "8" /  
274 "{" /  
275 "K" /  
276 "E" /  
277 "4" /  
278 "M" /  
279 "q" /  
280 "|" /  
281 "J" /  
282 "0" /  
283 "<" /  
284 "Z" /  
285 "e" /  
286 " " /  
287 "V" /  
288 "H" /  
289 "m" /  
290 "G" /  
291 "A" /  
292 "d" /  
293 "y" /  
294 "T" /  
295 "." /  
296 "S" /  
297 "j" /  
298 "I" /  
299 "#" /  
300 "J" /  
301 "p" /  
302 "\\\" /  
303 "F" /  
304 "b" /  
305 "'" /  
306 "W" /  
307 "k" /  
308 ";" /  
309 "5" /  
310 "N" /  
311 "s" /  
312 "2" /  
313 "1" /  
314 "1" /  
315 ":" /  
316 "r" /  
317 "-" /  
318 "`" /  
319 "h" /  
320 ">" /  
321 "t" /  
322 "}" /  
323 "^" /  
324 "+" /  
325 "U" /  
326 "B" /

```
327     "!" ,
328     "6" ,
329     ")" ,
330     "," ,
331     "P" ,
332     "g" ,
333     "=" ,
334     "C" ,
335     "D" ,
336     "?" ,
337     "R" ,
338     "9" ,
339     "z" ,
340     "i" ,
341     "n" ,
342     "w" ,
343     "f" ,
344     "%" ,
345     "u" ,
346 )
347 KEY_FOR_MAC_3 = (
348     "h" ,
349     "R" ,
350     "-" ,
351     "I" ,
352     "4" ,
353     "$" ,
354     "Z" ,
355     "c" ,
356     "/" ,
357     "Q" ,
358     "O" ,
359     "a" ,
360     "E" ,
361     "V" ,
362     " " ,
363     "q" ,
364     "A" ,
365     "!" ,
366     "," ,
367     "X" ,
368     "N" ,
369     "H" ,
370     "`" ,
371     "." ,
372     "K" ,
373     "e" ,
374     "*" ,
375     "v" ,
376     "\\" ,
377     "T" ,
378     "k" ,
379     "M" ,
380     "Y" ,
381     "^" ,
382     "6" ,
383     "l" ,
384     "j" ,
385     "G" ,
386     "i" ,
387     "m" ,
388     "x" ,
389     "7" ,
390     "3" ,
391     "L" ,
392     ";" ,
393     "C" ,
```

```
394 "1" /
395 "p" /
396 "2" /
397 "s" /
398 "]" /
399 "r" /
400 "%" /
401 "g" /
402 "@" /
403 "0" /
404 ")" /
405 "}" /
406 "B" /
407 "W" /
408 "|" /
409 "t" /
410 "=" /
411 "?" /
412 ">" /
413 "{" /
414 "b" /
415 "d" /
416 "!" /
417 "u" /
418 "f" /
419 "+" /
420 "P" /
421 ":" /
422 "z" /
423 "i" /
424 "E" /
425 "~" /
426 "[" /
427 "(" /
428 "&" /
429 "o" /
430 "n" /
431 "D" /
432 "#" /
433 "5" /
434 "8" /
435 "J" /
436 "w" /
437 "s" /
438 "U" /
439 "9" /
440 "<" /
441 "y" /
442 )
443 KEY_FOR_MAC_4 = (
444 " " /
445 "(" /
446 "H" /
447 "7" /
448 "~" /
449 "t" /
450 "F" /
451 "D" /
452 "j" /
453 "5" /
454 "C" /
455 "#" /
456 "o" /
457 "^" /
458 "X" /
459 "z" /
460 "9" /
```

461 "-" /  
462 "k" /  
463 " ' " /  
464 "N" /  
465 "E" /  
466 "+" /  
467 "q" /  
468 "P" /  
469 "/" /  
470 ";" /  
471 "S" /  
472 "W" /  
473 "h" /  
474 "s" /  
475 "f" /  
476 "." /  
477 "A" /  
478 "4" /  
479 " ' " /  
480 "\\\" /  
481 "Q" /  
482 "," /  
483 "m" /  
484 "e" /  
485 "\$" /  
486 "6" /  
487 "p" /  
488 "i" /  
489 "Y" /  
490 "|" /  
491 "U" /  
492 "a" /  
493 "3" /  
494 "L" /  
495 "g" /  
496 "c" /  
497 "]" /  
498 "0" /  
499 ")" /  
500 "v" /  
501 "x" /  
502 "8" /  
503 "T" /  
504 "l" /  
505 "u" /  
506 "b" /  
507 "[" /  
508 "V" /  
509 ":" /  
510 "M" /  
511 "r" /  
512 "B" /  
513 "G" /  
514 "1" /  
515 "K" /  
516 "d" /  
517 ">" /  
518 "O" /  
519 "`" /  
520 "\*" /  
521 "I" /  
522 "?" /  
523 "2" /  
524 "}" /  
525 "n" /  
526 "@" /  
527 "<" /



```
528     "&" /
529     "Z" /
530     "{" /
531     "R" /
532     "%" /
533     "!" /
534     "Y" /
535     "w" /
536     "=" /
537     "J" /
538 )
539 KEY_FOR_MAC_5 = (
540     "W" /
541     "f" /
542     "v" /
543     "l" /
544     "i" /
545     "h" /
546     "@" /
547     "C" /
548     ";" /
549     "!" /
550     "a" /
551     "8" /
552     "5" /
553     "S" /
554     "G" /
555     "X" /
556     "-" /
557     "{" /
558     "^" /
559     "~" /
560     "|" /
561     "m" /
562     "=" /
563     "&" /
564     "H" /
565     "O" /
566     "M" /
567     "Q" /
568     "0" /
569     "d" /
570     "I" /
571     "g" /
572     "6" /
573     "J" /
574     "9" /
575     "E" /
576     "z" /
577     "K" /
578     "L" /
579     "?" /
580     "/" /
581     ">" /
582     "o" /
583     "]" /
584     "n" /
585     "Y" /
586     "F" /
587     "t" /
588     "i" /
589     "D" /
590     "w" /
591     "#" /
592     "(" /
593     "`" /
594     "y" /
```

```
595     "%"/
596     "r"/
597     "["/
598     "p"/
599     "*" /
600     "b"/
601     "j"/
602     "<"/
603     "}" /
604     "i"/
605     "\\"/
606     "," /
607     "B"/
608     "R"/
609     "q"/
610     "U"/
611     "x"/
612     ")" /
613     "$"/
614     " " /
615     "1"/
616     "+" /
617     "T"/
618     "N"/
619     "s"/
620     "7"/
621     "c"/
622     "k"/
623     ":" /
624     "4"/
625     "3"/
626     "e"/
627     "z"/
628     "v"/
629     "P"/
630     "2"/
631     "." /
632     "u"/
633     "A"/
634 )
635 KEY_FOR_MAC_6 = (
636     "}" /
637     "X"/
638     "j"/
639     "%"/
640     "I"/
641     "T"/
642     "L"/
643     "s"/
644     "^"/
645     "m"/
646     "d"/
647     "K"/
648     ")" /
649     "-" /
650     "B"/
651     "D"/
652     "o"/
653     "E"/
654     "@"/
655     "Z"/
656     "`"/
657     "2"/
658     "f"/
659     "a"/
660     "6"/
661     "q"/
```

662 "e" /  
663 "n" /  
664 "\*" /  
665 "R" /  
666 "[" /  
667 ";" /  
668 "v" /  
669 "i" /  
670 "z" /  
671 "y" /  
672 "+" /  
673 "S" /  
674 "W" /  
675 "7" /  
676 "x" /  
677 "&" /  
678 "<" /  
679 "9" /  
680 "U" /  
681 "N" /  
682 "]" /  
683 "#" /  
684 "=" /  
685 "b" /  
686 "Q" /  
687 "Y" /  
688 "(" /  
689 "r" /  
690 "\\\" /  
691 "M" /  
692 "A" /  
693 "v" /  
694 "H" /  
695 "{" /  
696 "w" /  
697 "p" /  
698 "3" /  
699 "," /  
700 "k" /  
701 "i" /  
702 "t" /  
703 ":" /  
704 "u" /  
705 "J" /  
706 "8" /  
707 "F" /  
708 ">" /  
709 "i" /  
710 "G" /  
711 "P" /  
712 "0" /  
713 "/" /  
714 "~" /  
715 "|" /  
716 "h" /  
717 "O" /  
718 "g" /  
719 "-" /  
720 "c" /  
721 "l" /  
722 "C" /  
723 "4" /  
724 "?" /  
725 "1" /  
726 "." /  
727 "!" /  
728 "\$" /

```
729         "5",
730     )
731     KEY_FOR_MAC_7 = (
732         "U",
733         "F",
734         ".",
735         "T",
736         "n",
737         "}",
738         ")",
739         "E",
740         "I",
741         "S",
742         "{",
743         "\\ ",
744         ">",
745         "B",
746         "a",
747         "j",
748         "Z",
749         "/",
750         "<",
751         "P",
752         "e",
753         "O",
754         "|",
755         ",",
756         "i",
757         "0",
758         "1",
759         "6",
760         "@",
761         ";",
762         "X",
763         "M",
764         "J",
765         "#",
766         "2",
767         "7",
768         "+",
769         "s",
770         "]",
771         "h",
772         "x",
773         "&",
774         "Y",
775         "-",
776         "`",
777         "G",
778         "3",
779         ":",
780         "(",
781         "L",
782         "=",
783         "f",
784         "*",
785         "H",
786         "Q",
787         "R",
788         "8",
789         "4",
790         "z",
791         "[",
792         "^",
793         "5",
794         "g",
795         "D",
```

```
796     "Y" /
797     "!" /
798     "W" /
799     "$" /
800     "q" /
801     "9" /
802     "K" /
803     "k" /
804     " " /
805     "d" /
806     "p" /
807     "A" /
808     "u" /
809     "w" /
810     "C" /
811     "_" /
812     "o" /
813     "b" /
814     "i" /
815     "r" /
816     "t" /
817     "N" /
818     "v" /
819     "?" /
820     "m" /
821     "l" /
822     "c" /
823     "%" /
824     "V" /
825     "~" /
826 )
827 KEY_FOR_MAC_8 = (
828     "u" /
829     "5" /
830     " " /
831     "Y" /
832     "l" /
833     "p" /
834     "D" /
835     "L" /
836     "b" /
837     "k" /
838     "@" /
839     "9" /
840     "O" /
841     "0" /
842     "*" /
843     "}" /
844     "f" /
845     "w" /
846     "C" /
847     "d" /
848     " " /
849     "g" /
850     "t" /
851     "K" /
852     "&" /
853     ";" /
854     "s" /
855     "v" /
856     "o" /
857     "=" /
858     "7" /
859     "I" /
860     "]" /
861     "." /
862     "R" /
```

```
863     "c" /
864     "?" /
865     "a" /
866     "^" /
867     "v" /
868     "x" /
869     "%" /
870     "n" /
871     "\\\" /
872     "z" /
873     "B" /
874     "i" /
875     "Q" /
876     "U" /
877     "+" /
878     "m" /
879     "l" /
880     "," /
881     "F" /
882     "M" /
883     "$" /
884     "T" /
885     "3" /
886     "N" /
887     ")" /
888     "r" /
889     "y" /
890     ">" /
891     "8" /
892     "P" /
893     "q" /
894     "2" /
895     "<" /
896     "|" /
897     "`" /
898     "4" /
899     "~" /
900     "/" /
901     "A" /
902     ":" /
903     "J" /
904     "{" /
905     "(" /
906     "H" /
907     "-" /
908     "h" /
909     "i" /
910     "j" /
911     "G" /
912     "S" /
913     "X" /
914     "[" /
915     "6" /
916     "Z" /
917     "E" /
918     "W" /
919     "#" /
920     "e" /
921     "!" /
922 )
923 KEY_FOR_MAC_9 = (
924     "," /
925     "T" /
926     "g" /
927     "B" /
928     "5" /
929     "n" /
```

930 "f" /  
931 "s" /  
932 "G" /  
933 "W" /  
934 "V" /  
935 " ` " /  
936 "M" /  
937 "[ " /  
938 "!" /  
939 "e" /  
940 "L" /  
941 " \_ " /  
942 " ¯ " /  
943 "Q̄" /  
944 "9" /  
945 " ^ " /  
946 ") " /  
947 "P" /  
948 "0" /  
949 "j" /  
950 "2" /  
951 "v" /  
952 "?" /  
953 "E" /  
954 "Y" /  
955 "Z" /  
956 "1" /  
957 "| " /  
958 ". " /  
959 "#" /  
960 "w" /  
961 "D" /  
962 "J" /  
963 "l" /  
964 "O" /  
965 "@" /  
966 "t" /  
967 "{" /  
968 "x" /  
969 "+" /  
970 "4" /  
971 "S" /  
972 ";" /  
973 "u" /  
974 "F" /  
975 "~" /  
976 "q" /  
977 "}" /  
978 "<" /  
979 "N" /  
980 "o" /  
981 "(" /  
982 "K" /  
983 "/" /  
984 "z" /  
985 "a" /  
986 "\\ " /  
987 "p" /  
988 "3" /  
989 " i " /  
990 "i" /  
991 "8" /  
992 "7" /  
993 "%" /  
994 "6" /  
995 "U" /  
996 ">" /

```
997      "C" /
998      "X" /
999      "$" /
1000     "R" /
1001     "&" /
1002     "d" /
1003     "c" /
1004     "I" /
1005     "y" /
1006     "H" /
1007     "=" /
1008     " " /
1009     "h" /
1010     "r" /
1011     "m" /
1012     "*" /
1013     "b" /
1014     ":" /
1015     "A" /
1016     "]" /
1017     "k" /
1018 )
1019 KEY_FOR_MAC_10 = (
1020     "0" /
1021     "R" /
1022     "h" /
1023     "C" /
1024     "~" /
1025     "o" /
1026     "i" /
1027     "g" /
1028     "," /
1029     "k" /
1030     "8" /
1031     "H" /
1032     "Q" /
1033     "p" /
1034     "9" /
1035     "v" /
1036     "$" /
1037     "u" /
1038     "5" /
1039     "-" /
1040     "a" /
1041     "{" /
1042     "'" /
1043     "s" /
1044     "A" /
1045     ";" /
1046     "[" /
1047     "Y" /
1048     "^" /
1049     "r" /
1050     "M" /
1051     "b" /
1052     "G" /
1053     "_" /
1054     "m" /
1055     "e" /
1056     "x" /
1057     "I" /
1058     "T" /
1059     "n" /
1060     ">" /
1061     "<" /
1062     "P" /
1063     "J" /
```



```
1064     "U" ,
1065     "2" ,
1066     "3" ,
1067     "4" ,
1068     "#" ,
1069     "W" ,
1070     ":" ,
1071     "w" ,
1072     "D" ,
1073     "|" ,
1074     "(" ,
1075     "z" ,
1076     "q" ,
1077     "K" ,
1078     "V" ,
1079     "B" ,
1080     "L" ,
1081     "j" ,
1082     "d" ,
1083     "&" ,
1084     "!" ,
1085     "1" ,
1086     "+" ,
1087     "\\\" ,
1088     "X" ,
1089     "/" ,
1090     "?" ,
1091     "@" ,
1092     "*" ,
1093     "t" ,
1094     "c" ,
1095     "}" ,
1096     "Z" ,
1097     "%" ,
1098     "." ,
1099     ")" ,
1100     "E" ,
1101     "`" ,
1102     "f" ,
1103     "N" ,
1104     "y" ,
1105     "6" ,
1106     "7" ,
1107     "O" ,
1108     "=" ,
1109     "]" ,
1110     "F" ,
1111     "l" ,
1112     "S" ,
1113     "I" ,
1114 )
1115 KEY_FOR_MAC_11 = (
1116     "S" ,
1117     "}" ,
1118     "!" ,
1119     "N" ,
1120     "\\\" ,
1121     "C" ,
1122     "M" ,
1123     "[" ,
1124     "z" ,
1125     "?" ,
1126     "%" ,
1127     "7" ,
1128     "q" ,
1129     "Y" ,
1130     "." ,
```

1131	"\$"/
1132	"l"/
1133	"D"/
1134	"G"/
1135	"f"/
1136	"+"/
1137	"/"/
1138	"6"/
1139	"#"/
1140	"5"/
1141	"w"/
1142	"P"/
1143	"O"/
1144	"U"/
1145	"*"/
1146	"X"/
1147	"d"/
1148	"="/
1149	"3"/
1150	": "/
1151	"i"/
1152	"Y"/
1153	"h"/
1154	"v"/
1155	"("/
1156	"R"/
1157	"W"/
1158	"x"/
1159	"'"/
1160	"1"/
1161	"c"/
1162	"A"/
1163	"<"/
1164	"J"/
1165	" "/
1166	"L"/
1167	" "/
1168	"T"/
1169	"^"/
1170	">"/
1171	"H"/
1172	"2"/
1173	"`"/
1174	"I"/
1175	"/"/
1176	"0"/
1177	"k"/
1178	"-"/
1179	"u"/
1180	"o"/
1181	"&"/
1182	"Z"/
1183	"j"/
1184	"p"/
1185	")"/
1186	"/"/
1187	"r"/
1188	"9"/
1189	"'"/
1190	"Q"/
1191	"4"/
1192	"]"/
1193	"m"/
1194	"t"/
1195	"B"/
1196	"a"/
1197	"F"/

```

1198     "v",
1199     "b",
1200     "8",
1201     "E",
1202     "n",
1203     "g",
1204     "@",
1205     "e",
1206     "s",
1207     "~",
1208     "{",
1209     "K",
1210 )
1211
1212
1213 # !The end of the characters list and key for mac encryption layers
1214
1215
1216 # !---The start for the all encryption and decryption functions---
1217
1218
1219 def reverser(text):
1220     """This function reverses the text"""
1221     return text[::-1]
1222
1223
1224 # ---The end of reverser function.---
1225
1226
1227 def mac_encode(text, key):
1228     """Generic Mono-Alphabetic cipher encoder with given key."""
1229     result = []
1230     for char in text:
1231         if char in CHARACTERS:
1232             index = CHARACTERS.index(char)
1233             result.append(key[index])
1234         else:
1235             result.append(char)
1236     return "".join(result)
1237
1238
1239 def mac_decode(text, key):
1240     """Generic Mono-Alphabetic cipher decoder with given key."""
1241     result = []
1242     for char in text:
1243         if char in key:
1244             index = key.index(char)
1245             result.append(CHARACTERS[index])
1246         else:
1247             result.append(char)
1248     return "".join(result)
1249
1250
1251 def all_mac_encryption(text):
1252     """12 Layers combined MAC encryption"""
1253     text = mac_encode(text, KEY_FOR_MAC_5)
1254     text = reverser(text)
1255     text = mac_encode(text, KEY_FOR_MAC_4)
1256     text = mac_encode(text, KEY_FOR_MAC_2)
1257     text = mac_encode(text, KEY_FOR_MAC_1)
1258     text = mac_encode(text, KEY_FOR_MAC_3)
1259     text = mac_encode(text, KEY_FOR_MAC_8)
1260     text = mac_encode(text, KEY_FOR_MAC_7)
1261     text = mac_encode(text, KEY_FOR_MAC_9)
1262     text = mac_encode(text, KEY_FOR_MAC_6)
1263     text = mac_encode(text, KEY_FOR_MAC_11)
1264     return mac_encode(text, KEY_FOR_MAC_10)

```

```

1265
1266
1267 def all_mac_decryption(text):
1268     """12 Layers combined MAC decryption"""
1269     text = mac_decode(text, KEY_FOR_MAC_10)
1270     text = mac_decode(text, KEY_FOR_MAC_11)
1271     text = mac_decode(text, KEY_FOR_MAC_6)
1272     text = mac_decode(text, KEY_FOR_MAC_9)
1273     text = mac_decode(text, KEY_FOR_MAC_7)
1274     text = mac_decode(text, KEY_FOR_MAC_8)
1275     text = mac_decode(text, KEY_FOR_MAC_3)
1276     text = mac_decode(text, KEY_FOR_MAC_1)
1277     text = mac_decode(text, KEY_FOR_MAC_2)
1278     text = mac_decode(text, KEY_FOR_MAC_4)
1279     text = reverser(text)
1280     return mac_decode(text, KEY_FOR_MAC_5)
1281
1282
1283 def encrypt(text: str) -> str:
1284     """Doubled 12 Layer MAC encryption + compression"""
1285     encrypted = all_mac_encryption(all_mac_encryption(text))
1286     compressed = compress(encrypted.encode("utf-8"))
1287     return b64encode(compressed).decode("utf-8")
1288
1289
1290 def decrypt(text: str) -> str:
1291     """Doubled 12 Layer MAC decryption + decompression"""
1292     compressed = b64decode(text.encode("utf-8"))
1293     encrypted = decompress(compressed).decode("utf-8")
1294     return all_mac_decryption(all_mac_decryption(encrypted))
1295
1296
1297 # !---The sending and reading messages functions---
1298
1299
1300 def send_message_non(username, message):
1301     msg_e = encrypt(
1302         f"{datetime.now().strftime('%Y-%m-%d %H:%M')}:{username}: {message}"
1303     )
1304     msg.delete("1.0", "end")
1305     ftp_host = FTP_HOST.get()
1306     ftp_user = FTP_USER.get()
1307     ftp_pass = FTP_PASS.get()
1308     chat_name = f"{chat_file_name.get()}.txt"
1309     local_temp = "chat_temp.txt"
1310
1311     try:
1312         try:
1313             ftp = FTP_TLS(ftp_host)
1314             ftp.login(ftp_user, ftp_pass)
1315         except Exception:
1316             ftp = FTP(ftp_host)
1317             ftp.login(ftp_user, ftp_pass)
1318
1319         try:
1320             with open(local_temp, "wb") as f:
1321                 ftp.retrbinary(f"RETR " + chat_name, f.write)
1322         except Exception:
1323             pass
1324
1325         with open(local_temp, "w", encoding="utf-8") as f:
1326             f.write(f"{msg_e}\n")
1327
1328         with open(local_temp, "rb") as f:
1329             ftp.storbinary(f"APPE " + chat_name, f)
1330
1331         ftp.quit()

```

```

1332         read_messages()
1333
1334     except Exception as error:
1335         messagebox.showerror("Error sending message", str(error))
1336
1337
1338 def send_messages(username, message):
1339     Thread(target=send_message_non, args=(
1340         username, message), daemon=True).start()
1341
1342
1343 def read_messages_non():
1344     ftp_host = FTP_HOST.get()
1345     ftp_user = FTP_USER.get()
1346     ftp_pass = FTP_PASS.get()
1347     chat_name = f"{chat_file_name.get()}.txt"
1348     local_temp = "chat_temp.txt"
1349
1350     try:
1351         try:
1352             ftp = FTP_TLS(ftp_host)
1353             ftp.login(ftp_user, ftp_pass)
1354         except Exception as e:
1355             ftp = FTP(ftp_host)
1356             ftp.login(ftp_user, ftp_pass)
1357
1358         with open(local_temp, "wb") as file:
1359             ftp.retrbinary(f"RETR {chat_name}", file.write)
1360             ftp.quit()
1361
1362         with open(local_temp, "r", encoding="utf-8") as file:
1363             lines = file.readlines()
1364
1365         out_lines = []
1366         for line in lines:
1367             try:
1368                 out_lines.append(decrypt(line.strip()))
1369             except Exception:
1370                 out_lines.append(line.strip())
1371
1372         message_text = "\n".join(
1373             out_lines) if out_lines else "No messages yet."
1374
1375         chat_notes.config(state=NORMAL)
1376         chat_notes.delete("1.0", "end")
1377         chat_notes.insert("end", message_text)
1378         chat_notes.see("end")
1379         chat_notes.config(state=DISABLED)
1380
1381     except Exception as error:
1382         messagebox.showerror("Error reading messages", str(error))
1383
1384
1385 def read_messages():
1386     Thread(target=read_messages_non, daemon=True).start()
1387
1388
1389 def auto_refresh_func():
1390     if FTP_HOST.get() and FTP_USER.get() and FTP_PASS.get() and chat_file_name.get():
1391         read_messages()
1392         main.after(5000, auto_refresh_func)
1393     else:
1394         messagebox.showwarning("Warning", "Please, fill in all details")
1395
1396
1397 def start_loop():
1398     send_butt.config(state=NORMAL)

```

```

1399     auto_refresh_func()
1400     done_butt.config(state=DISABLED)
1401     done_butt.config(text="Loop Started")
1402
1403
1404     main = Window(themename="darkly")
1405     main.title("TLock")
1406     main.geometry("1920x1080")
1407     main.state("zoomed")
1408     menubar = Menu(main)
1409     main.config(menu=menubar)
1410
1411     # Menubar
1412     accessibility_menu = Menu(menubar, tearoff=False)
1413     accessibility_menu.add_command(label="Help", command=help_func)
1414     accessibility_menu.add_command(label="Exit", command=lambda: main.destroy())
1415     menubar.add_cascade(label="Accessibility", menu=accessibility_menu)
1416
1417     # Widgets
1418
1419     FTP_HOST = StringVar()
1420     Label(main, text="Host:").place(x=5, y=5)
1421     Entry(main, textvariable=FTP_HOST, width=225).place(x=100, y=5)
1422
1423     FTP_USER = StringVar()
1424     Label(main, text="FTP User:").place(x=5, y=40)
1425     Entry(main, textvariable=FTP_USER, width=225).place(x=100, y=40)
1426
1427     FTP_PASS = StringVar()
1428     Label(main, text="Password:").place(x=5, y=75)
1429     Entry(main, textvariable=FTP_PASS, width=225, show="*").place(x=100, y=75)
1430
1431     username_var = StringVar()
1432     Label(main, text="Username:").place(x=5, y=110)
1433     Entry(main, textvariable=username_var, width=225).place(x=100, y=110)
1434
1435     chat_file_name = StringVar()
1436     Label(main, text="Chat File:").place(x=5, y=146)
1437     Entry(main, textvariable=chat_file_name, width=201).place(x=100, y=146)
1438
1439     done_butt = Button(
1440         main,
1441         text="Done",
1442         command=lambda: start_loop(),
1443         width=20,
1444         bootstyle="primary",
1445     )
1446     done_butt.place(x=1730, y=146)
1447
1448     Label(main, text="Chat:").place(x=5, y=180)
1449     chat_notes = ScrolledText(main, width=233, height=16)
1450     chat_notes.place(x=5, y=210)
1451     chat_notes.see("end")
1452     chat_notes.config(state=DISABLED)
1453
1454     Label(main, text="Message:").place(x=5, y=547)
1455     msg = ScrolledText(main, width=233, height=16)
1456     msg.place(x=5, y=575)
1457
1458     send_butt = Button(
1459         main,
1460         text="Send",
1461         command=lambda: send_messages(username_var.get(), msg.get("1.0", "end")),
1462         width=235,
1463         bootstyle="primary",
1464         state=DISABLED
1465     ).place(x=5, y=920)

```

```
1466
1467  main.mainloop()
1468
```