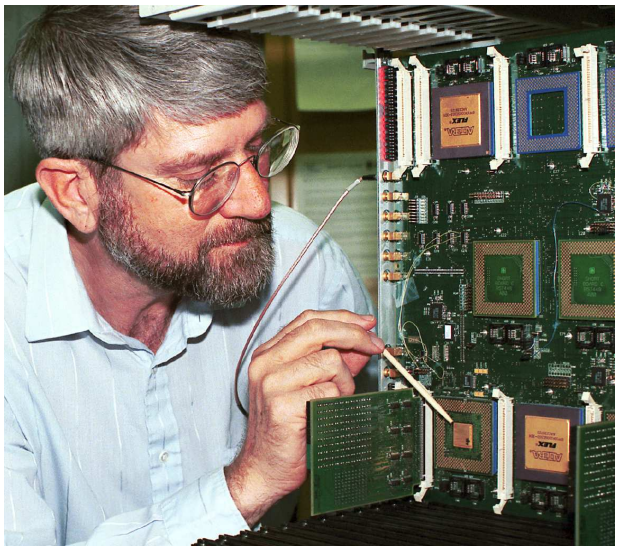
EECS 281: Extra Credit (35 points) #3 **Due: Tuesday, November 23, 2004**



Sandia researchers develop world's fastest encryptor

ALBUQUERQUE, N.M. – The world's fastest encryption device, developed at the Department of Energy's (DOE) Sandia National Laboratories, should soon be protecting data being transmitted from supercomputers, workstations, telephones and video terminals. It encrypts data at more than 6.7 billion bits per second, 10 times faster than any other known encryptor.

Currently, the fastest commercial encryptor operates at 0.15 billion bits per second, which means long waits for large amounts of data to move from supercomputers to visualization stations, for example. The DES ASIC is the first encryption device fast enough to secure the standard 2.5 Gb/s and 10 Gb/s communication channels now being used to carry the ever increasing data traffic for Internet commerce. See www.sandia.gov

This assignment combines history, technology and business: **Telecipher systems** were developed during the First World War by AT&T Bell Labs engineer, Gilbert Vernam (1890 - 1960), in 1918, and co-invented the one-time pad cipher. The machine scrambled 5-bit Baudot codes (pre-ASCII codes) using exclusive-or operations. In the 1940s, Claude Shannon, also at Bell Labs, proved that the one-time pad is unbreakable. It is the first and only encryption method for which there is such a proof.

The **Lorenz "Schlüsselzusatz 40"** Telecipher machine was adopted by the German High Command for high-level communications in World War II. "Schlüsselzusatz" meant "cipher key attachment" and meant that the device connected was a box without a keyboard or printer that was used in conjunction with a conventional teletype-writer.

Figure 1: Schlüsselzusatz 40

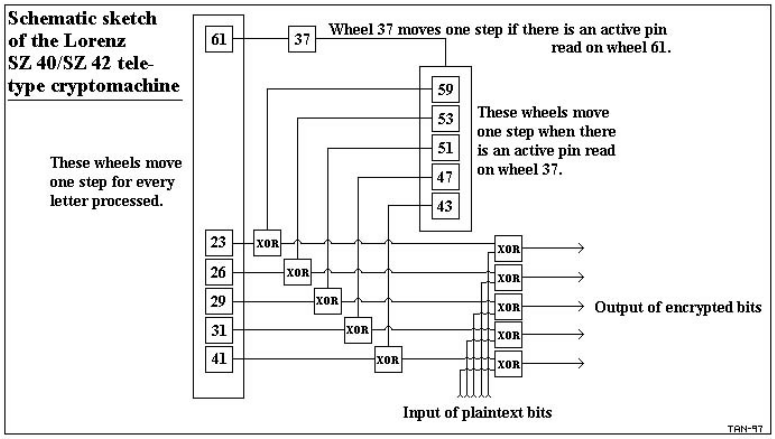


Figure 2: SZ-40 Wiring diagram

The British cryptographers of Bletchley Park code-named German teleprinter ciphers FISH because the code breakers in Bletchley Park never actually saw Lorenz machine until right at the end of the war even though they had been breaking the Lorenz cipher for two and a half years. The Bletchley Park estate had been a manor since the Norman invasion and was the site of a secret British military intelligence operations during World War II. In 1943, Max Newman and his team built Colossus, the world's first electronic digital programmable computer, to crack FISH cipher messages. The first "re-programmable" computer was the "Electronic Delay Storage Automatic Calculator (EDSAC)" and was built in 1949 by Maurice Wilkes of Cambridge University in the England.

Encryption chipset website	NASDAQ symbol
http://www.sonicwall.com/General/DisplayDetails.asp?id=10	SNWL
http://www.mykotronx.com/products/ASIC	SFNT (owned by http://www.safenet-inc.com)
http://www.ocean-logic.com/products.htm	SNPS (acquired by Synopsys)

(1) (5 points extra credit) Lookup the following companies and determine why they still need to use a hardware-based solution instead of software for encryption and **(b)** list applications their chips are needed for? **(c)** Give the stock-chart comparisons on http://finance.yahoo.com. Would you buy their stock? and why?

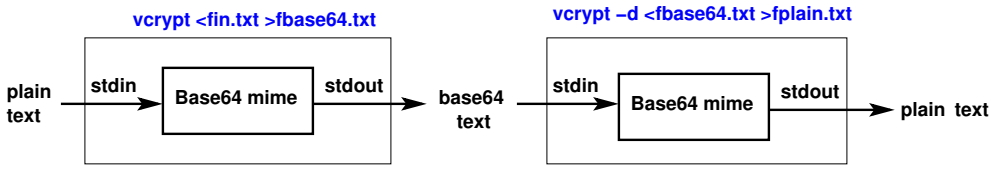


Figure 3: For problem 2, vcrypt base64 mime flow

(2) (10 points extra credit) Write a program which as shown in Figure 3 in standard ANSI C/C++ language which reads in a plain text ASCII file from stdin and outputs a base64 mime file passed into the command line (i.e. msdos command prompt). The `-d` option will decode a mime base64 file into a plain ASCII text file. Figure 6 shows the skeleton sample code in order to get started. You must re-write strcpyBase64, fputcBase64 and fgetcBase64 to really handle the mime base4 character processing.

Using stdin and stdout allows the same vcrypt program to do both convert and decode via the pipe operator "|" and test your program: "`vcrypt <fin.txt | vcrypt -d >fplain.txt`". The "`diff fin.txt fplain.txt`" command will allow you to compare the two files: "`vcrypt <fin.txt | vcrypt -d | diff - fin.txt`".

Unix commands for windows XP can be downloaded from "http://www.simtel.net". To verify your vcrypt program, sample files will be located on "http://bear.ces.cwru.edu" website. Hand in a paper copy as well as email a zipped file to the grader rxv20@case.edu. The grader may require a demo from you.

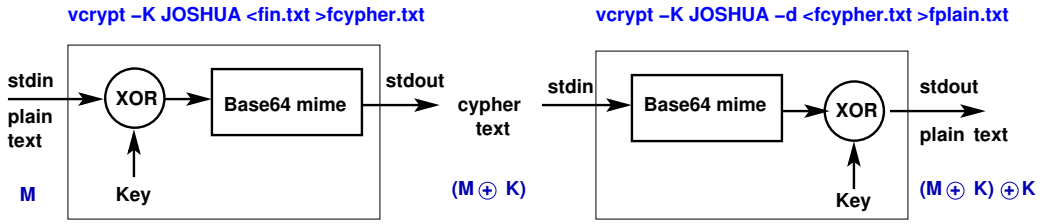


Figure 4: For problem 3, vcrypt message flow

(3a) (10 points extra credit) Using the XOR theorems given on the last exam, prove that $M = ((M \oplus K) \oplus K)$.
(3b) Extend the vcrypt command from part 2 as shown in Figure 4. The vcrypt command now outputs an encrypted stdout file in base64 by XORing the base64 key with the plain text. The command line key is in base64 and convert to binary before applying it to the file. If no key is given then the default key is eight characters in length and is zero. By using a zero key, is this upward compatible with part 2? The `-d` option will decrypt a base64 file into a plain ASCII text file.

In the 1983 film *WarGames*, the actor Matthew Broderick uses the password, JOSHUA, to gain access to NORAD. Test your program the following ways: "`vcrypt -K JOSHUA <fin.txt | vcrypt -d -K JOSHUA >fplain.txt`" and also, "`vcrypt -K JOSHUA <fin.txt | vcrypt -d -K JOSHUA | diff - fin.txt`".
(4) (10 points extra credit) In cryptanalysis, a **crib** is a sample of known plaintext. The word was adapted from the student term meaning a bit of a cheat, 'I cribbed my answer from your test paper'. A **known-plaintext attack** is

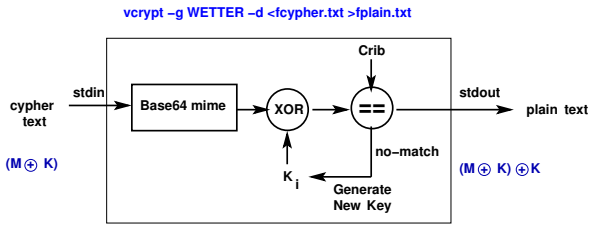


Figure 5: For problem 4, vcrypt cracking

a cryptanalytic attack in which the attacker has samples of both the plaintext and its encrypted version and makes use of them to reveal the secret key. Crib examples include stereotyped salutations, endings, titles, routing codes, etc. For example, typical German messages began with the phrase "WETTER" (in English, "weather report").

(4a) Add a crib option (i.e. -g WETTER) in order to crack an encrypted file that has an unknown key. Use the -K option as a key seed, if -K is not entered then assume the default of problem 3. The procedure is as follows: (a) copy seed key to the trial key, (b) read in the same number of characters as the crib, (c) apply the key to the read in characters, (d) if result matches with the crib then decrypt using this key and output to stderr the discovered key in base64, (d) else increment the key by one and goto step (c).

(4b) If the key is 8 characters, what is the maximum keys to try?

```
/* ------------------------------ Figure 6: Skeleton C code ------------------------------ */
int fputcBase64(int c, FILE *stream) { /* Writes four base64 characters on every third fputcBase64 call */
  static int ncalls=0, buffer[4]; /* static variable keeps the value on exit and re-entry */
  return fputc(c, stream); /* if c is EOF then flush any un-written characters */
}
int fgetcBase64(FILE *stream) { /* Reads four characters from file, on every third fgetcBase64 call */
  static int ncalls=0, buffer[4]; /* ..re-write.. */ return fgetc(stream); /* return original ASCII character */
}
int strcpyBase64(char *binary, char *base64) { /* (used only in problem 3) convert a base64 string into binary bytes */
  strcpy(binary, base64); /* re-write this code */
  return strlen(binary); /* return the number of converted bytes */
}
int keyinc(char *key, int keyn) { /* increment the key by 1 (used only in problem 4) */
  return 0; /* return 1 if incremented beyond key size */
}
int main(int argc, char *argv[]) {
  int  i, c, keyn, dflag=0; char key[256]; FILE *fin=stdin; FILE *fout=stdout;

  for(i=1; i<argc; i++) {
    if      (!strcmp(argv[i], "-K")) { keyn=strcpyBase64(key, argv[i++]); } /* (used only in problem 3) */
    else if (!strcmp(argv[i], "-d")) { dflag=1; } /* decode/decrypted */
    else {
      fprintf(stderr, "vcrypt: unknown keyword[%s]\n\n", argv[i]);
      fprintf(stderr, "vcrypt -K base64key [-d] <fin.txt >fout.txt\n"); exit(1);
  } }
  i=0;  c=fgetc(fin);
  while(c!=EOF) {
    c= c ^ key[i++]; if (i>keyn) { i=0; }
    fputcBase64(c, fout); c=fgetc(fin);
  }
  fputcBase64(EOF, fout); /* flush any un-written characters */
}
```

references: **The Lorenz Cipher and how Bletchley Park broke it:** http://www.codesandciphers.org.uk/lorenz/fish.htm

Teleprinter ciphers, U.S. WWI system, SZ40: http://hem.passagen.se/tan01/tele.html

Telecipher systems with a detailed example: http://www.vectorsite.net/ttcode9.html

Sneakers, DVD, 1992, "Setec Astronomy" is a anagram for "Too many secrets". They're hired by companies to try to break into them, notably their computer systems, or "sneaking". http://www.imdb.com/

Enigma, DVD, 2001, About the Brits at Bletchley Park who broke the German U-boat communications code.