~~SECRET~~

NATIONAL SECURITY AGENCY
FORT GEORGE G. MEADE, MARYLAND

# CRYPTOLOG

## FEB-MAR 1984



~~HANDLE VIA COMINT CHANNELS ONLY~~

P.L. 86-36

~~SECRET~~

~~CLASSIFIED BY NSA/CSSM 123-2
DECLASSIFY ON: Originating Agency's Determination Required~~

# CRYPTOLOG

Published by P1, Techniques and Standards

VOL. XI, No. 2-3 FEBRUARY-MARCH 1984



For subscriptions
send name and organization
to: [redacted] P14

P.L. 86-36

To submit articles or letters
by mail, to: P1, Cryptolog

via PLATFORM mail, send to:
cryptolg at barlcØ5
(bar-one-c-zero-five)
(note: no 'O' in 'log')



## Editorial

*As an agency, we live today on yesterday's discoveries. Today's output, which pays the bills around here, is based largely upon technical breakthroughs made sometime in the past. Most of our people are working to produce today's results, but here and there, mostly in back rooms, there are a few scattered people doing the "discovery" work. We used to call them "break-in artists." They are busy making tomorrow's production possible and, in a very real sense, making it possible for tomorrow's bills to be paid.*

*Once in a while, one runs across a whole cluster of this discovery work. It is as if a renaissance had broken out in one particular shop. A whole group of people seem to be bubbling over with invention, intuition, and discovery. It is an exciting place to be, when it happens.*

*There used to be one or two managers who seemed to have such a renaissance around them wherever they went. They seemed to have the knack of creating an atmosphere that fostered discovery, that encouraged breakthroughs.*

*I can remember studying those managers, to see if I could emulate their evident ability to stimulate the discovery process. I can remember going to management courses and reading various books on the latest fads in management styles, looking for clues about how to generate the atmosphere that discovery and creativity seem to need. I can't remember finding much that was useful; it seemed to be easier to talk about things that were easier to count or measure.*

*For an outfit that depends so much on break-in artists, we ought to worry about finding, growing, and managing tomorrow's crop. Perhaps we already are.*

[redacted]

# ADMIRAL A.I. NEPENIN: FATHER OF MODERN RUSSIAN NAVAL INTELLIGENCE (U)

P.L. 86-36

by

[redacted]

*Originally prepared as an Appendix to the author's article on "Communications Intelligence and Tsarist Russia," which appeared in the Jan 84 issue of Cryptolog.*

The history of Russian military affairs has been one of incompetence mixed with flashes of brilliance. The brilliance has usually been in the form of individual military "shakers and movers" who have risen to the occasion with determination and forcefulness to carry through their goals, come what may, to the end. One might include Marshals Suvorov and Zhukov or Admirals Senyavin and Gorshkov in this category. However, there is one individual, although he is little known in the West, who as a "shaker and mover" might be said to be the father of modern Russian naval intelligence: Admiral Adrian Ivanovich Nepenin.

Nepenin, in his capacity as Chief of the Baltic Fleet's Communications (and Intelligence) Service both prior to and during World War I, built the naval intelligence organization into a formidable arm of the Russian Navy and ultimately established roots which have carried over into the Soviet era.

Adrian Ivanovich Nepenin was born 21 October 1871 in Pskov Province, Russia. He entered the Russian Naval Academy in 1885 and graduated in 1889. In 1898 he was assigned to the Far East Fleet. In December 1904 Captain 2nd Rank Nepenin was assigned to command the destroyer STOROZHEVOJ at Port Arthur. During the war with Japan, Nepenin was captured and spent the last part of that war as a POW in Japan. Between 1905 and 1910 Nepenin held various ship commands in the Baltic Fleet.

In 1910, after much thought, Nepenin sent a plan for reorganization of the Communications and Observation Service of the Baltic Fleet to Admiral Nikolaj Ottovich von Ehssen, Commander-in-Chief, Baltic Fleet. Admiral von Ehssen liked Nepenin's energetic idea for the Communications Service and in 1911 appointed Nepenin as Chief of the Communications Service. Nepenin probably made Captain 1st Rank at this time.

Over the next few years, under Nepenin's guidance and direction, the Communications Service--almost alone within the Russian Navy --achieved a high esprit de corps among all its personnel. By October 1915 Nepenin had achieved the rank of Rear Admiral for his efforts. His admirers included not only his own men but even foreign allies assigned to Russia during the war. During a visit to a Communications Service airbase in the Baltic in 1916, Admiral Sir Richard Phillimore (British Naval Representative to Russian General Staff Headquarters, "STAVKA," 1915-16) was quoted as telling the Communications Service officers and men:

"Everything is excellent in our British Navy ... except that we do not have such an Admiral as your Nepenin who knows everything."[1]

On 6 September 1916, largely on the basis of his Communications Service record, Nepenin was offered and accepted the command of the Baltic Fleet along with the rank of Vice Admiral. Nepenin's time as CINC, however, was brief with little opportunity to carry out his ideas on reorganizing and revitalizing the spirit of the Fleet. On 15 March 1917, while on his way to meet with a group of disgruntles sailors near the Helsingfors Railway Station, Nepenin was killed by a shot from behind by either a mutinous sailor (according to the Soviet version) or a German agent dressed in the uniform of a Baltic Fleet sailor (Russian emigré version).[2]

Although Nepenin's period on the stage of History was brief, he left an indelible imprint on the development of Russian naval intelligence in the 20th century.

FOOTNOTES

1. Apparently Sir Richard forgot (?) in his remarks about Admiral Reginald "Blinker" Hall of "Room 40 OB" fame.

2. Dudorov, Rear Admiral Boris Petrovich Dudorov in the emigré journal Morskie Zapiski (The Naval Records), New York. See also The Russian Navy in War and Revolution by G. K. Graf, Munich: R. Oldenburg, 1923, pp. 119-121, and The Russians at Sea by David Woodward, London: William Kimber, 1965, pp. 181-182. For the traditional Soviet negative view of Nepenin from 1916 as "suppressor of the Revolutionary in the Baltic Fleet," see Pavlovich, N. B. (editor), Flot v Pervoj Mirovoj Vojne (The Navy in World War I), 2 vols, Moscow: Voenizdat, 1964, Vol 1, p. 241.

# THE CRYPT BUG (U)

by [redacted]

P.L. 86-36

When all good folks are sound asleep,
And all the rest are counting sheep,
He concentrates on cipher text,
And contemplates ways most complex
To render an approved solution
Of some obscure substitution.

While all the world is sleeping, snoring
Loud enough to rip the flooring,
He derives much satisfaction
From the spatial interaction
Of poly-graphic frequencies
And isomorphic sequences,
Of characters on paper slips
Better know as sliding strips.

Slides them West and tries the "Chi" test,
Slides them East and tries the "Phi" test,
Clamps his pipe tight in his mouth,
And grimly slides them North and South,
And if success eludes him then,
Tears them up and starts again.
Meanwhile the clock ticks on and on,
Until at long last comes the dawn.

As the milkman rattles by,
He is heard to heave a sigh,
Slowly piles the work sheets higher,
Calmly throws them on the fire,
Having proved one simple fact;
There can be do doubt of that--
As suspected all along,
Everything he did was wrong.

from Signal Corps Bulletin No. 109,
July-December 1940

***Human Factors***

# USER-FRIENDLY WRITING (U)

[redacted] P13 P.L. 86-36

We have seen and heard a lot lately about our writing. Our Director has made a special point of urging us to write more clearly and directly. A hard-hitting article on the same topic may be found in the November 1983 issue of CRYPTOLOG, pp. 13-18. A number of services are available to help us improve our communication skills, including courses at the School and the new "Write-Line." The quality and effectiveness of our writing and speaking is far more important than many of us seem to realize, in spite of these management initiatives. Unfortunately, our writing will only get better if we care about it and feel that it matters. I am not going to launch into a long article about good writing, or how to improve our writing. That has been done already by many others; I will mention two sources that I have found particularly useful. But I feel that good clear writing is an important human factors issue, and I'd like to say a few things about it in these Tech Notes.

I read a lot of technical papers and research reports, and I edit my office's Monthly Research Summaries. I am sorry to say that I have seen a great deal of very bad writing. It is bad because it is not "user-friendly." I am going to direct my comments to anyone out there who writes the kinds of prose I have to fight my way through each month in our Research Summary.

As a reader, I am a user of your paper or report, just like a user of any other tool. The paper probably says something I need to know or I wouldn't have picked it up. If you create long, intricate sentences choked with jargon, you are putting major obstacles in my way. You are making me spend far too much of my time and energy to get your meaning. Sometimes your sentences are so complicated that you lose your own way through them, so how can you expect me, the reader, to understand them? I know that you don't set out to mystify the reader on purpose. I believe that scientific and technical writers have certain basic misconceptions about writing; some or all of these they probably learn from their teachers at colleges and technical schools, many of whom are also apallingly bad writers. Let's take a look at some of the faulty assumptions that may give rise to the bad writing technical people so often produce.

"If I say it simply, people will think I'm uneducated."

People in technical fields have gotten so used to a certain very heavy, convoluted style of writing that simpler writing just sounds inappropriate and anticlimactic to them. Even if they are just telling us that they debugged a program or checked out some minor electronic gadget, they feel they must sound like a candidate for the Nobel prize.

"If I say it simply, people won't know it's important."

Many people seem to think that the length of their words and the complexity of their sentences are a direct measure of the importance of the topic. I "use" a Kleenex to blow my nose, but I "utilize" the computer, because the computer is a lot more expensive and important than a Kleenex or my nose! I might "make it easier" for the cat to use the litter box, but I feel I must "facilitate user accessibility" to project X.

"If I say it simply, I won't be able to hedge and fudge."

Technical and scientific people are masters of the art of hedging their bets. To some extent, this is necessary and justified; we have a professional obligation to specify the degree of significance of a result, the reliability of a statement, or the statistical context of an event. We have to convey these matters to our readers at those times and places where they are important and appropriate. Unfortunately, the hedging gets to be a habit, so that it infects all our writing, and shows up in lots of places where it serves no purpose. I suspect that the long sentences starting out with endless strings of subordinate clauses arise in this hedging habit. Each subordinate clause is like a safe little fence to push the bald, direct subject and verb further away from the reader, until the meaning disappears in a comfortable mist. I have seen some cases where the subject and main verb never arrive at all. In many cases, the writer has forgotten whether the subject was singular or plural, or even what the subject started out to be, by the time he gets to the main verb. It's a real help to the reader when you put the main subject and verb at or near the beginning of the sentence. Don't get into the habit of writing English as if it were German!

A frequent error I see in technical writing is the "dangling participle." The long string of subordinate clauses at the beginning of the sentence often starts with a participial phrase that does not refer to the real subject of the sentence. Strunk and White (reference 2 below) say, "A participial phrase at the beginning of a sentence must refer to the grammatical subject." [p. 8] As the reference states, sentences violating this rule are often ludicrous, for example, "Being in a dilapidated condition, I was able to buy the house very cheap." Even when they aren't ridiculous, dangling participles are confusing and sloppy. This kind of writing doesn't impress a careful reader with the quality of the writer's thinking.

"My readers are all experts in my field and know the jargon."

Perhaps this is true; if so, I think the writer is making a mistake. What about managers in other organizations that might make use of his ideas? They may be familiar with the field at a global level without knowing all the buzzwords and abbreviations he tosses off in his report. What about technical people in related fields? They may have a similar problem with some of the jargon. Finally, I maintain that jargon and alphabet soup are far too often a lazy substitute for thinking. If we understand what we are doing, we should be able to express it clearly with a minimum of jargon. When I am talking to someone who throws a lot of alphabet soup and jargon at me, I make a point of asking politely for one or two definitions or expansions. Very often, I get a blank look, a silence, then "Well, gosh, now that you ask, I don't know!"

"Oh, EVERYBODY knows what that means!"

The remarks in paragraph 4 apply to this one too. I came across the phrase "repartitioning the functionality" in a recent research summary. I very much doubt that "everybody" knows what that might mean, and I'm sure that some simpler, clearer way could have been found to express the idea, whatever it was.

"If I simply say 'somebody did thus and so,' I am leaving somebody's posterior alarmingly uncovered."

We seem to think it is much safer for all concerned to use the passive voice. Nobody DID it. It just happened. It was done. That also sounds much more impressive, like an act of God: it rained, there was light. We've also had it hammered into us throughout a technical or scientific education that we must always be "objective." The worst sin in the world is to be "personal" or "subjective"! That's another reason why we avoid the active voice like the plague and prefer passives or impersonal constructions like "there were indications that" and "it is apparent that." These constructions make our sentences needlessly complicated right at the start: harder for us to write, and harder for the reader to read. At their worst, they can totally obscure the meaning.

Here's a sample of user-unfriendly prose to illustrate the needless syntactic tangles and sloppy semantics of bad writing: "In addition to examining the use of, and designing a gadget for a frammus for project GLITCH, the use of a widget for project FOO was also studied." Exercise: find the subject of this sentence. Here's a better way of saying it: "We designed a gadget for a frammus for project GLITCH, and examined its use. We also studied the use of a widget for project FOO." I am still unhappy about the vagueness of "studying the use" of gadgets and widgets. Does the writer mean "try out the gadget to see how useful it is"? Or does he mean "observe operators using the gadget and study how they use it"? Maybe he means "perform various experiments to see if there is any point in trying to use the gadget." When we look closely at this sentence, we see that it doesn't convey much meaning to the reader unless he already knows all the intimate details of the projects and equipment.

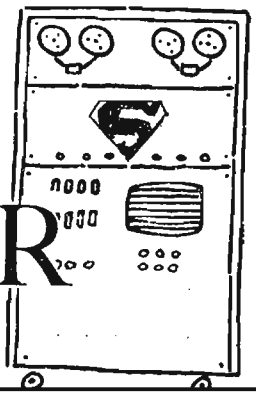
In closing, I'd like to stress one final point: writing matters. It matters HOW something is expressed. Engineers and mathematicians know that the formal systems they use (mathematical and scientific notation, models, and methods) are powerful tools. Computer systems people hold up certain standards for writing good code and for the efficient, economical use of programming languages. Technical people respect those tools and appreciate the value of elegance and economy in their use. Natural language is another tool, just as powerful and deserving of respect. Unfortunately, too many technical and scientific workers tend to ignore or look down on natural language. They don't think of English as a tool that can and should be used with elegance and skill. Their mathematics may be beautiful, and their programs may be clear and economical, but if their writing is messy their minds are likely to be a bit messy too. The exercise of stating something clearly and directly in good plain English can often clear up the mess for the writer as well as his readers.

References

"Just Plain English," Department of English, US Air Force Academy, Colorado 80840 (no date).

Strunk, W., Jr., and E. B. White, The Elements of Style, New York, Macmillan, 1972

# NATIONAL SUPERCOMPUTER RESEARCH CENTER(U)



b

P.L. 86-36

## Introduction

(U) A National Supercomputing Research Center is important to NSA because it will help us to solve many future supercomputing problems. The word "supercomputing" simply means the intelligent use of the most powerful computational tools currently available. Such a center will probably solve these problems better than we have done before and in a way to help other national defense efforts as well. It will do this with outside people and outside money. But we need to fight for it.

## Background

(U) The Chief Scientist of NSA, Mr. Kermith Speierman, was asked by DIRNSA to formulate NSA recommendations for DoD regarding supercomputer initiatives. The Speierman Committee was formed to develop those recommendations and reported to the Director in the autumn of 1983, urging four functions for a federal supercomputing initiative to help supercomputing:

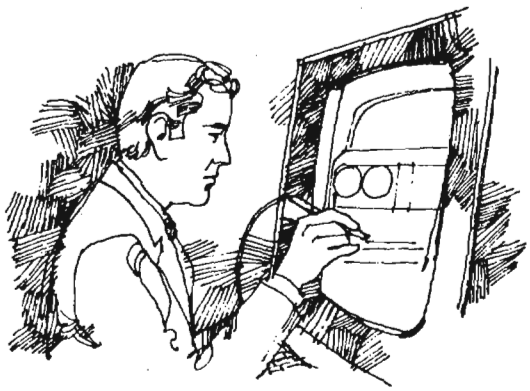
a. In-house, NSA: Highly classified special projects;

b. Defense Parallel Processing Laboratory (DPPL): Medium-level classified work on massively parallel processing for national security in the next decade;

c. NSRC: A largely unclassified lab for supercomputing hardware and software research, with special emphasis on support of:

d. Regional Computational Facilities (RCFs): An unclassified program to provide supercomputer access to academic researchers.

(U) The in-house function is already being performed and will continue. If no other initiatives are acted on, RCFs will be partially done by the National Science Foundation (NSF) and the Department of Energy (DoE) laboratories under existing plans. The really new features are the DPPL and NSRC. But the DPPL seems to be on its way to receiving acceptance. Therefore, this article is dedicated solely to justifying the NSRC.



Possible Objections to the NSRC

(U) The major objections to a new, independent NSRC are four:

1. No need because of current open research;
2. The DoE labs could do this (and they want to);
3. An intense, open research program would transfer information and technology to the outer world; and
4. Suggestions for an NSRC would arouse opposition from DoE or the President's Office of Science and Technology Policy (OSTP) and thus possibly imperil the whole initiative.

(U) I believe that objections 1 and 2 are essentially false (as stated) and that 3 and 4 are true but can still be handled.

Objection 1

(U) This objection is that no radically new efforts in unclassified supercomputing research are necessary because of existing work in government, industry, and academia. However, a look at specific examples (e.g., operating systems and software) shows how inadequate the current efforts really are. The vendors typically supply poor operating systems and FORTRAN. After all, operational software is not their main interest and something really sophisticated is quite beyond their current capability. The result is that the users either get substandard performance from their machines or have to develop new operating systems and languages, usually different from anybody else's.

(U) The DoE labs have developed their own operating systems with a line editor and complicated user commands that would be unsuitable for NSA. The NSA supercomputing environment--i.e., the [redacted] system and IMP language--is powerful and easy to use. Yet it cannot be the general supercomputing standard for various technical reasons. In addition, it is difficult to transfer to different machines. If we soon have a wide variety of supercomputers, it will be impossible for us to maintain [redacted] IMP on all without a great increase in the number of systems programmers. UNIX/C may become the de facto standard since it will soon be available on almost all supercomputers. However, we see it as having inherent inefficiencies that make it difficult to use the full power of the computer when we wish to.

P.L. 86-36

(U) One possible response is to put this problem in the DPPL or keep it in NSA (by using more people). But the systems programming problem is essentially unclassified. How much better to free up NSAers and DPPLers for classified work and put systems software in the NSRC, where it will be serving an independent need anyway (support of the regional centers). Driven by a variety of applications from academia, with a few clever interns from the labs and NSA bringing the best of their methods, the NSRC could have a resounding success. Specifically, they might well develop once and for all a portable, easy, powerful environment that could be used by all and enhance the vendors' products at the same time. And the really great thing is the leverage we get by having this work done by other people with others' money. Similar statements could surely be made in the other areas of NSRC emphasis besides languages and operating systems; i.e., algorithms, hardware technology, architecture, numerical analysis, artificial intelligence, and graphics.

Objection 2

(U) Los Alamos National Labs would dearly love to have the functions of the NSRC. However, even a casual glance at their record must produce skepticism inasmuch as:

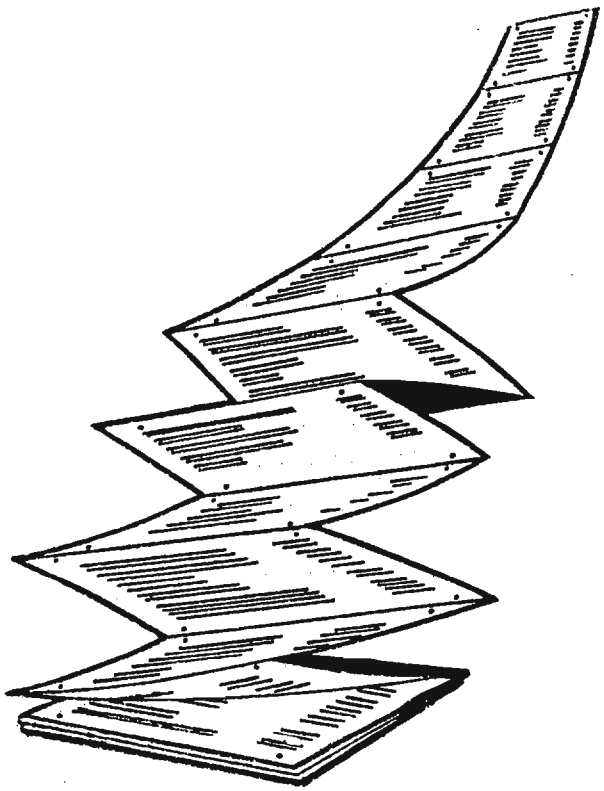
[] they get relatively poor performance from their Crays (the current standard supercomputers);

[] they have a clumsy operating system;

[] they discourage assembly language and modern high-level languages; and

[] they have relatively few experts, partly because they have not encouraged (as NSA has) scientific personnel to become relatively sophisticated.

(U) Maybe they will change if the labels on their doors are changed, but I doubt it. And I doubt that even "safeguards" written into new terms of reference, or even a change of location, would really change their modus operandi. If Los Alamos gets the NSRC, then I predict that the whole effort will be irrelevant to NSA and we will be back to having to use many NSAers and DPPLers to do unclassified work.

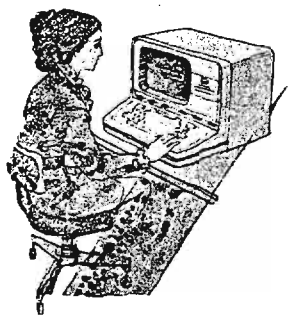


Objection 3

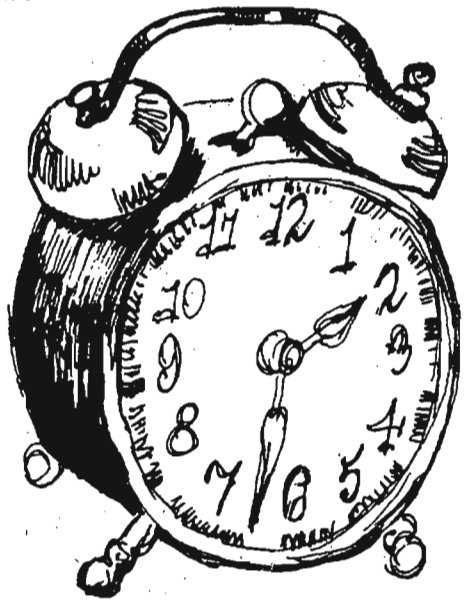
(U) The Speierman federal initiative would result in some information transfer to the outside. However, since the outside world is no longer very far behind us, the real question is what will be the marginal increase in harm (as opposed to what would happen anyway), weighed against the potential benefits to us. Since the in-house programming and the DPPL are classified, the only threat comes from the regional centers and the NSRC. The regional centers should provide only computational access at the end of a telephone line, and that only by grant. Thus the foreign graduate student in astrophysics could get time to study galactic structure, but he could not dump critical software, and he would have to break the terms of his grant to study cryptography on the sly. The NSRC itself should be physically restricted to US nationals since it will have at least company proprietary, and possibly classified, information. The problem with the NSRC is that useful hardware and software work will eventually become public. After all, the people there will be developing very powerful unclassified operating systems. My contention is that the outside world is catching up anyway. It is far better to have them trying to get up to the level of our unclassified base a few years after us than for us to have an unclassified base behind that of other countries and to try to build our classified technology from it.

Objection 4

(U) If the NSRC is worth having, it's worth fighting for. We should not regard it as a political chip to be bargained away for DoE support for the whole initiative. The best approach is to keep trying to persuade the interested parties, especially DoE, that the NSRC is in their best interest too. They also will get leverage from having the NSRC solve their problems.



# SHELL GAME



## TIME SHELLS [U]

**by WES**

You may not have noticed, but the time function on our UNIX systems has been converted to GMT, or ZULU time. The other day, the phone rang and the voice at the other end said, "The boss would like to see you at 2:45 today." Since I was on the system and probably would be for most of the day, I typed in

```
remind 2:30
See boss at 2:45
```

and finished with a control-D. Then, being a cautious sort (remind has sometimes had a mind of its own), I typed in 'delrem' and looked at what the system thought it was going to do. By now you have guessed that the system, operating in the time zone of the mythical kingdom of ZULU, had stored away my "wake up call" as 1430Z. So much for modern efficiency.

Now I don't really mind using ZULU time, but it's just three more things to remember: the summer difference, the winter difference, and which are we in right now. Frankly, I'm still trying to remember all my PIN numbers (how many bank cards do you have?), and all the passwords to the various systems, and a couple of door combinations, and...well you get the idea. Every time I get another one of these important things to remember, I forget something trivial like a birthday or an anniversary.

So I went looking for some way to get the system to keep track for me. What I found were two shells, one short and sweet, and the other much more involved. Here is the first one, called "tyme":

```
date "+%H" | = t
expr $t - 05 | = t
date "+TIME: %H:%M:%S ZULU ($t:%M EST)"
```

What I hadn't realized was how much the 'date' program had changed since UNIX Version 6. Since Daylight Saving Time runs from the last Sunday of April to the last Sunday of October, I added some commands and the shell now looks like this:

```
date "+%m" | = a
date "+%d" | = b
date "+%w" | = c
expr $b - $c | = d
switch "$a"
:       'standard time'
: 11
: 12
: 01
: 02
: 03
        = e S
        = f 05
        breaksw
:       'last Sunday in April change'
: 04
        if $d -ge 24 then
                = e D
                = f 04
                breaksw
        else
                = e S
                = f 05
                breaksw
        endif
:       'daylight saving time'
: 05
: 06
: 07
: 08
```

```
: 09
        = e D
        = f 04
        breaksw
:       'last Sunday in October change'
: 10
        if $d -ge 25 then
                = e S
                = f 05
                breaksw
        else
                = e D
                = f 04
                breaksw
        endif
endsw
date "+%H" | = t
expr $t - $f | = t
date "+TIME: %H:%M:%S ZULU ($t:%M E$eT)"
```

At the other end of the scale, I found the shell 'time1', written by [redacted] in P14. It begins in the following column.

P.L. 86-36

The original version of Bob's shell uses reverse video to set up a rather startling display on the screen. It will also clobber your terminal if you try to use it across the network. If you get the original version, you could insert a test to see whether the terminal of the user was a network terminal, something like:

```
switch "$t"
: [X-Z]
        (change to net-friendly version ...)
        ...
endsw
```

depending upon how the network terminals are labelled on your host. Then all you need is a second version of those lines that have reverse video, replacing them with whatever your artistic heart desires.

After some discussion, we decided to print the shell without the inverse video, in the interests of minimizing the chaos around the TSS community.

```
goto start
: Bob Jones, P14, 3369-s
: (5741-s)-- 04 Mar 83
: See list of variables at end of file
: start
date "+%H" | = t
date "+%d" | = d
date "+%j" | = c
expr $t - 5 | = l
expr $t + 3 | = m
expr $t + 09 | = k
expr $t + 11 | = f
if $k -gt 23 then
expr $k - 24 | = k
expr $d + 01 | = a
expr $c + 1 | = b
else
expr $d + 0 | = a
expr $c + 0 | = b
endif
if $f -gt 23 then
expr $f - 24 | = f
expr $d + 01 | = g
expr $c + 1 | = h
goto skip
else
expr $c + 0 | = h
= g "$d"
expr $g + 0 | = g
endif
: skip
if "$g" -lt "10" then
= g "0$g"
else
endif
if "$d" -lt "10" then
= d "0$d"
else
endif
if "$h" -lt "100" then
= h "0$h"
else
endif
if "$b" -lt "100" then
= b "0$b"
else
endif
if "$f" -lt "10" then
= f "0$f"
else
endif
if "$k" -lt "10" then
= k "0$k"
else
endif
if "$a" -lt "10" then
= a "0$a"
else
endif
if "$l" -lt "10" then
= l "0$l"
else
endif
if "$m" -lt "10" then
= m "0$m"
```

```
else goto rundate
endif
: rundate
pump
^G
```

```
**************************************************************************
**************************************************************************
**                                                                      **
**                                                                      **
!
date "+** LOCAL---  DATE: %d %h %y  TIME: $l:%M (EST) JULIAN DATE:  %y%j **"
echo "**                                                                 **"
date "+** ZULU----  DATE: %d %h %y  TIME: $t:%M (Z)   JULIAN DATE:  %y%j **"
echo "**                                                                 **"
date "+** MOSCOW--  DATE: %d %h %y  TIME: $m:%M (C)   JULIAN DATE:  %y%j **"
echo "**                                                                 **"
date "+** KOREA---  DATE: $a %h %y  TIME: $k:%M (I)   JULIAN DATE:  %y$b **"
echo "**                                                                 **"
date "+** FIJI----  DATE: $g %h %y  TIME: $f:%M (L)   JULIAN DATE:  %y$h **"
echo "**                                                                 **"
pump
**                                                                      **
**************************************************************************
**************************************************************************
```

```
^G
!
exit
: 'VARIABLES-(refered to as $t, $m, etc)--t or $t=system hour; d or $d=system '
: 'date; c=system Julian Day; l=local time; m=Moscow time; k=Korean time; '
: 'f=Fiji Time; The following are computed if the time is after 2400 -- '
: 'a=Korean Date; b=Korean J=Day; g=Fiji Day; and h=Fiji J=day. '
: 'Other computations such as 'if $m -lt "10" then' place a zero in front of '
: ' '$m'.  This, and the statements such as 'if "$h" -lt "100" then' are '
: 'required because the math functions will drop leading zeros. '
: ' ^G    -- Rings Terminal Bell'
```

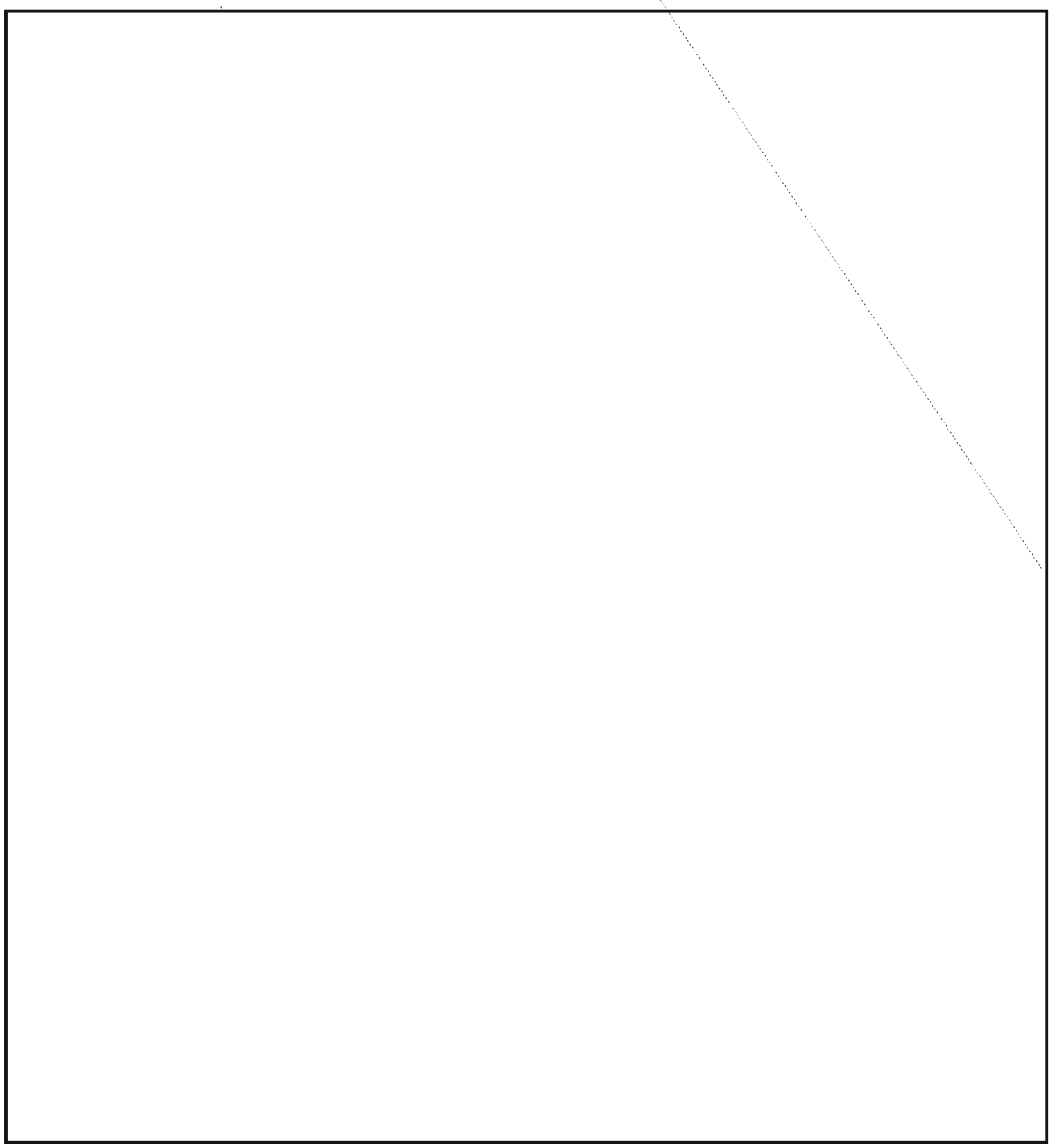
Bob also has a version of this that runs on the IBM PC in living color. I'm sure he would be happy to let you have a copy of either version.

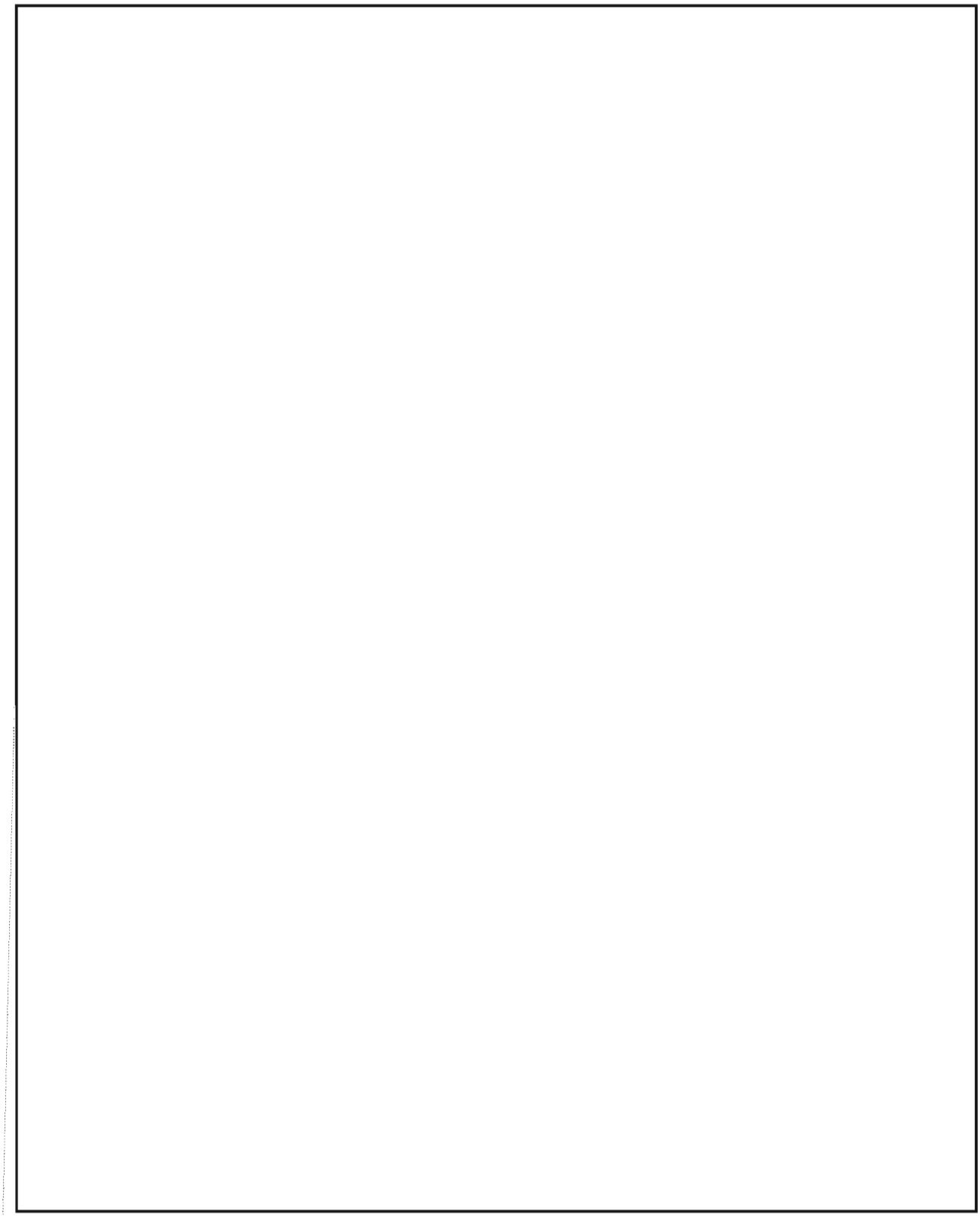
These shells are more for demonstration than anything else, and that is the spirit in which they are presented here. For example, the first shell does not add a leading zero when the local hour is less than ten, and will probably do something weird if the local hour is less than 5. The third shell doesn't quite understand what to do at the end of the month and the 31st day in the land of ZULU may become the 32nd in some other time zone. If some reader comes up with a good fix, we will be happy to print it.

P.L. 86-36

# NSA-Crostic #53

In case you were born on February 29th — Leap Year Day — well then, Happy Birthday to you, too!





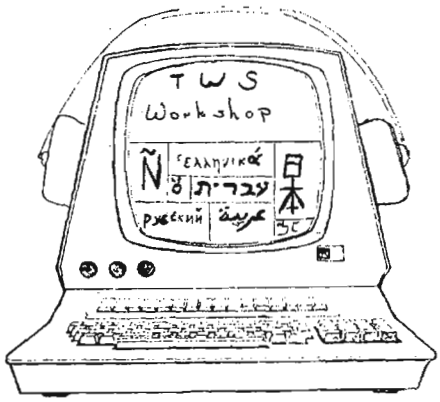
P.L. 86-36

# BE PART OF THE PROCESSing

## CRYPTO-LINGUISTIC ASSOCIATION

## LANGUAGE AUTOMATION COMMITTEE

## presents




# Translator/Transcriber Work Station

Are you now using computer power in your language activities?

Will you be using it soon?

Feeling frustrated, intimidated, or uninformed about language automation in your office?

At the TWS Work Shop you can
* learn about current and future computer systems
* express your ideas
* share your concerns

4 - 7 June 1984
2W087

| | |
|---|---|
| Monday, Tuesday, Wednesday | 0830-1100 |
| repeated at | 1300-1530 |
| Thursday Wrap-up | 1300-1500 |

All interested Green-Badge personnel invited

See you there!

# AUTOMATED INFORMATION SECURITY[U]

EDITED BY EDWIN F. STEEBLE, C22

P.L. 86-36

*This article is extracted from the Department of Defense Computer Security Center's (DoD CSC) response to the USMC. The Marines had requested computer security guidance and evaluations of several architectural plans. That paper was authored by [redacted] Chief of the Applications Evaluations Systems Office, with aid from [redacted] Chief Scientist, DoD Computer Security Center and [redacted] Col, USAF, Deputy Director, DoD Computer Security Center. The COMSEC policy, procedures, and guidance were supplied by [redacted] COMSEC Doctrine and Threat Assessment Office; [redacted] COMSEC Standards and Evaluations Office; and [redacted] Jr., COMSEC Applications Office.*

*For this publication minor editing and revisions, mostly to delete USMC specifics, were done by [redacted] Chief, Operational Systems Evaluation Division, DoD Computer Security Center.*

## I. Computer Security Guidance

### A. Policy Requirements

Computer security requirements derive from the need for the information processing system to control access to classified information. These requirements are described more fully in the DoD CSC Trusted Computer System Evaluation Criteria, 15 August 1983 [1]. Briefly, such systems are required to implement the following:

[] MARKING - An ADP system which is used to process or handle classified or other definitely categorized sensitive information shall clearly store and maintain the integrity of classification or other sensitivity marking labels for all information. The system shall assure that the classified or other sensitive information is accurately marked when included in output from the ADP system.

[] MANDATORY SECURITY - The computer system must enforce the formal system of information control reflected in the security classification designation and special handling restriction set associated with the sensitive information handled or processed by the ADP system together with the clearance set associated with the individuals who may request access to the information.

[] DISCRETIONARY SECURITY - The computer system must enforce access limitations placed on classified or other sensitive information based on identified individuals or groups of individuals who have been determined to have a Need-to-Know for the information.

[] ACCOUNTABILITY - An ADP system which is used to process or handle classified information must account for usage on a named-individual basis whenever classified information is generated or accessed.

[] CONTINUOUS PROTECTION - Security-relevant portions of a trusted computer system must be maintained under configuration control to assure that unauthorized changes have not been made which could possibly subvert the system's ability to control classified information.

These policy requirements form the basis for defining security requirements at the system level, as well as for the hardware and software components of the system. They also determine procedural requirements to support the continuous protection policy and assure the operational effectiveness of technical safeguards.

The degree to which a system must comply with these requirements, either in the use of specific security features or in the degree of assurance that the features are effective, is a function of risk of exploitation. This risk depends upon motivation, capability, and opportunity of an opponent to exploit the system's protection controls and mechanisms. These factors, in turn, are influenced by such things as the most sensitive information in the system, the least restrictive clearance of system users or those associated with its development and operation, the hostility of the environment, and time.

### B. System Requirements

A primary system requirement is to have a clearly defined security perimeter that includes a suitable combination of manual and automatic trusted processes to control access to classified or sensitive data in the system. Each such process is designed and operated to implement a well-defined interpretation of DoD security policy (e.g., minimally, information that is labeled SECRET will not be accessible by personnel holding less than a SECRET clearance). The perimeter may be entirely defined by environmental (i.e., physical, personnel, and operational security) controls, as is the case in a dedicated mode of operation. It may require hardware, software, and COMSEC controls in addition to the environmental controls. For example, electrically connecting two different computer systems requires hardware and software controls over the interfaces between systems operating at different system-high levels. These controls must ensure, for example, that the integrity of classification labels on internal files is protected and that information flowing from one system to another is classified no higher than the maximum authorized for the receiving system. This, in turn, requires assurance that the integrity of classification labels on internal files is protected in the computers. In the multilevel mode one relies very heavily on controls internal to the computer to enforce applicable security policy, and thus the computer hardware and software controls become an even more critical element of the security perimeter.

The specific security requirements, both technical and environmental, to be enforced by a computer systems application are prescribed by the Designated Approving Authority (DAA), in accordance with DoD Directive 5200.28 or DCI Computer Security Directive "Security of Intelligence Information in Automated Systems and Networks" (formerly DCID 1/16), while the requirements for determining the technical efficacy of the system's security controls and mechanisms are stated in the Center's Trusted Computer Systems Evaluation Criteria. The DAA is then required to make an explicit decision to use the system operationally when convinced that these security requirements are satisfactorily met. We elaborate below on the computer hardware/software certification and accreditation process to support this.

### C. Hardware Requirements

Computer systems that are trusted to enforce a security policy employ a combination of hardware and software mechanisms. The hardware mechanisms of concern are those that simplify and optimize the implementation of access control over the subjects and objects as defined in the formal security policy model abstraction. Below we list desirable features worth considering in the selection of a hardware architecture. Note that these features, while helpful, do not supplant the need for a security kernel. However, they may improve performance throughput significantly over the pure use of software controls.

[] Virtual Memory - This hardware feature is essential. It can be realized in either a page- or a segmented-based organization and would provide an effective environment for multiple processes. Both require address mapping circuitry that automatically provides access checking during address translation.

[] Execution Domain - It is minimally essential that the hardware support two execution domains (preferably three), where one domain is privileged and protected from the less privileged domain. Security kernel software runs within the most privileged domain, and untrusted user software executes within the less privileged domain(s).

[] Controlled Access to I/O Devices - It is essential that computer architecture provide some mechanism that enables a security kernel to maintain control over accesses to input/output (I/O) devices. A sufficient solution is the notion of

privileged I/O operations. Here, I/O is performed only by a process executing in the appropriate privileged domain. The kernel must control access to this privileged state.

[] Multiple Processes - Many users normally share concurrently the available resources of a general-purpose computer system, therefore the base computer architecture must provide support for an efficient multiple-process structure. The minimal hardware support necessary is the capability to save and restore process definition information.

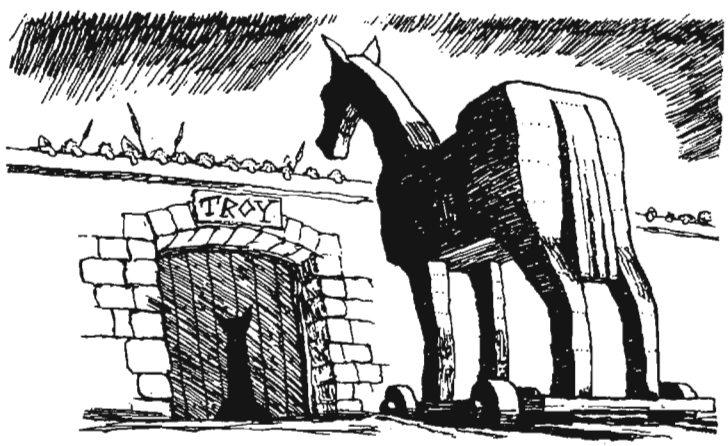
Additional information may be found in MITRE Technical Report No. ESD-TR-78-170 "Minicomputer Architectures For Effective Security Kernel Implementations" by John D. Tangney, dated October 1978.

Because of the reliance one has on these controls, there are several security concerns to be addressed in the acquisition and use of this hardware. One concern is correctness. Assurances must be given to show that the hardware mechanisms have been designed and built to function correctly. A second concern is reliability. Failures in the hardware must not weaken or eliminate the security controls that are implemented in the hardware itself or in the software which, in turn, requires correctly functioning hardware. A third concern is integrity. Configuration control measures during hardware design, implementation, operation, and maintenance must deter accidental or deliberate modifications of the hardware that can cause security controls to be bypassed or weakened. The degree of concern in each area and the corresponding steps taken to reduce the risk is application-dependent. Although exploiting such avenues of vulnerability is possible, one must consider them in the context of other areas which could be more susceptible to attack (e.g., software).

In those cases where the hardware will be used in a periods processing mode, it should permit rapid and reliable erasure of all internal memory (e.g., primary storage, non-removable secondary storage and buffers). It must also support the capability for a physical disconnect from those other devices in areas with a lesser degree of protection. There is ongoing research as part of the consolidated DoD Computer Security R&D program to develop a "job stream separator" which automatically and reliably performs all necessary color change procedures.

In those cases where the computer will simultaneously process or store information of different classifications, the hardware should support internal labeling of files with the appropriate security classification, and these internal labels should be used as the primary basis for access control decisions. This is particularly the case if the system users are not all authorized access to all of these files (e.g., as in the controlled or multilevel mode of operation). A similar requirement may exist for systems which process personnel proprietary or other sensitive unclassified information.

Individual hardware components must meet TEMPEST requirements commensurate with their operational environment, current policy, and the perceived threat of exploitation.



### D. Software Requirements

Software that must enforce DoD security policy must be designed, implemented, and documented to permit credible evaluation and verification that it, in fact, correctly enforces that policy. This requirement would have to be applied to all system software including the operating system, system utilities, data base management systems (DBMS), compilers, or application software. Such evaluation would be difficult and lack credibility if the security-relevant mechanisms are complex and scattered throughout the software. One simply cannot determine that an unstructured collection of these mechanisms correctly implements the policy and cannot be circumvented. Thus, the Center requires that in trusted computer systems all security-related functions be implemented in well-defined portions of software, firmware, and hardware, the totality of which is called the trusted computing base (TCB). The TCB must be designed and implemented so that its security controls are always invoked and are tamperproof, that is, the controls cannot be modified or bypassed by the remaining (untrusted) portions of the system and that they be of sufficiently simple design as to be subjected to thorough test and analysis. During its design and development, the TCB is subjected to specification and design analysis verification and testing to assure that these properties are indeed satisfied. The DoD CSC Trusted Computer System Evaluation Criteria amplify these requirements further.

Determining the specific requirements for software controls and level of assurance, i.e., the evaluation class, for a particular application must reflect the level of risk and degree of trust required of the hardware and software. One indicator of this is security range that is, the difference between the classification of the most sensitive information and the least restrictive user clearance. Thus, for example, a Class C2 system may provide adequate trust for a system-high application. A multilevel mode application would, on the other hand, normally be expected to meet the criteria of a Class B2 or higher system, depending on its security range.

### E. Procedures

The continuous protection requirement is primarily satisfied with procedures to control and monitor access to hardware and software security components during their design and implementation, and then during their operational life cycle. Such procedures are a critical part of gaining assurance that the security mechanisms are designed and built to meet stated requirements and then maintained and used to remain effective. Specific requirements include:

[] clearing system support personnel to the highest level of data in the system;

[] clearing maintenance personnel commensurate with the sensitivity of information to which they could get access; and

[] developing and maintaining software which protects sensitive information in an environment consistent with the sensitivity of the data being protected and with a level of risk that is acceptable to owners of sensitive information.

In systems which involve periods processing, accreditable procedures are needed to change processing classification levels. Procedures include removing sensitive data from the system, disconnecting or reconnecting peripheral devices and remote terminals, and rebooting the appropriate operating system at the new processing level.

### F. Classified Software

The security mechanisms and their implementation in trusted system hardware and software are generally unclassified. However, as noted earlier, this software may be treated as if it were classified to meet the continuous protection requirement. There may be instances in which security-related software is classified (e.g., if it implements a classified cryptographic algorithm) or security-related software contains classified data (e.g., the routing tables in a message system). Such software must be protected like any other classified information while it is stored in the computer. There may be multiple copies of it in primary and secondary storage, all of which must be labeled and protected, as must all hardcopy printouts of it.

G. General

It is DoD policy that all ADP systems which process classified information will be accredited; that is, there will be an explicit decision that the system adequately protects information and can be used operationally. This accreditation is frequently based upon a technical evaluation of the system to determine how well it meets predefined requirements. However, unless the system is designed and built to be evaluated, as is NOT the case with most existing computer systems, the technical evaluation consists almost entirely of looking for flaws in the system or conducting tests of the system's ability to withstand penetration. Neither case gives assurance that the system is secure because such exhaustive testing never finishes. Thus, it is vitally important that security requirements be identified early in the system's development. It is equally important that the system security architecture identify trustworthy mechanisms to control the flow of information into, out of, and within the system. One can then determine explicitly the policy model which each trusted hardware and software component of this architecture must enforce and the appropriate Trust Class as described in the Criteria. One can then specify, implement, verify, and certify that those enforcement mechanisms that are implemented correctly enforce the policy. To assist with this, there is a growing collection of formal design and verification methodologies which can be used. These include SRI's Hierarchical Development Methodology, University of Texas' GYPSY system, and SDC's Formal Development Methodology. The C organization is undertaking an effort to make these tools more easily available to and usable by system developers as well as by NSA and DoD system test and evaluation organizations.

Computer vendors, (i.e., DEC, UNIVAC, Honeywell, etc.) have developed or are developing trusted systems which might meet long-range requirements. Additionally, software houses are developing add-on packages to provide a little increase in software security (i.e., SKK's ACF2, IBM's RACF, CGA's Top Secret, etc.). In Section III below we note other possible uses of trusted systems as part of the security architecture. Thus, a first step in developing the architectural strategy and planning for using trusted systems would be to determine what the long-term security requirements are (i.e. will multilevel security become an operational necessity, and if so, over what range of classification and user clearance?).

ADP Security Certification/Accreditation Planning Guide (reference #2) provides additional information on the critical steps in the certification/accreditation process. Further direct interaction with the user, designer, and C2 could follow the reading of this literature and enable C2 to work on recommending or finalizing a recommended secure system.

II. Telecommunications

A well-defined, layered network security architecture is needed that

[] addresses all the threats of concern to the user; and

[] is consistent with, or is at least not incompatible with, the security architectures of networks to which various users are connecting.

It is desirable to have a single, layered, inter-network security architecture that can be deployed across all DoD certified nets. An ambitious DoD effort is under way to achieve this initiative.

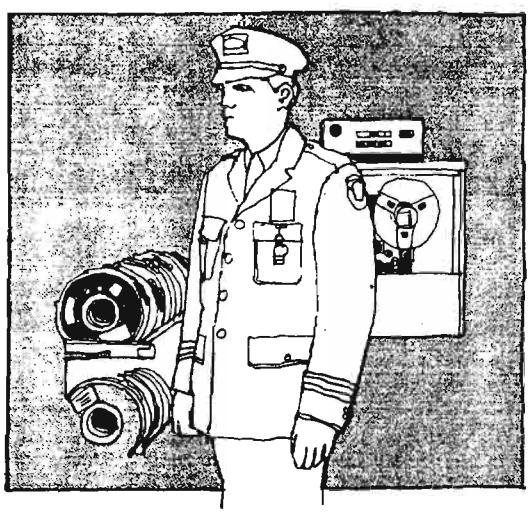
III. Policy

Electrical Interfaces - Electrical interfaces between systems operating at different classification levels must ensure that only appropriately classified information flows from the more sensitive to the less sensitive system. It must also prevent users of the less sensitive system from making unauthorized changes, accidentally or deliberately, to data in the other system or from disrupting its use. A manual interface has, until recently, been the accepted method. However, trustworthy devices for controlling such interfaces have been proposed for several systems. One such device currently in development will use the Honeywell SCOMP as a basis for implementing a GUARD to allow SECRET users to access SECRET data bases on the US Army Forces Command's Top Secret system-high WWMCCS computer. There is another approach which uses a cryptographically derived cryptographic check to verify the releasability of information when it is being electronically transferred between security perimeters (reference #3).

*-Property[2] - DoD security policy for ADP systems was discussed in Section I above. The *-property is one part of the Bell & La Padula[3] policy model for mandatory security. It is more conservative than DoD policy as it relates to paper documents but it precludes the success of Trojan Horse attacks.

Data Aggregation - DoD policy for correct classification and handling labels for data elements (alone or in aggregate) should be implemented in data processing systems. This requires reliable labels on internal files and on output giving the classifications or other special handling instructions, as determined by the owner of the information at the field, record, file, or data base level, as appropriate.

Data Encryption Standard (DES) - Present policy requires that NSA approve, on a case-by-case basis, any proposed use of DES to protect classified communications. With respect to the use of DES to protect unclassified, national security-related communications, recently issued national policy requires that Services, Departments, and Agencies determine the risk of exploitation of their unclassified communications, either in consultation with or based upon prior guidance from NSA in accordance with Federal Standard (FS) 1027. Where there is high risk of exploitation, NSA will prescribe or approve the cryptographic system used, on a case-by-case basis. For all other applications, commercial cryptographic systems (to include DES) may be used if they have been endorsed for general application by NSA.



## IV. General

There will be additional costs associated with implementing, using, and maintaining physical, emanations, personnel, and procedural security safeguards. Some of this additional cost (e.g., for physical and emanations safeguards) is part of the capital investment. On the other hand, the costs for personnel and procedural safeguards are part of the operational costs. The actual costs for a facility depend upon the level of protection required for the information being processed in a given threat environment. There will also be additional costs associated with acquiring and using trusted computer systems. Designing security into the system can lower these costs and have a beneficial payoff through improved reliability and maintainability which results from a well-structured software design and implementation. We note that there are two key aspects to be considered in estimating the cost of safeguards in these security areas. They are (1) what level of protection is required, and (2) how must these safeguards be used and maintained to ensure their continued effectiveness?

[Doesn't protection of sources, methods, and products require this? EFS]

### Footnotes

1. This and the other referenced papers can be obtained from the DoD CSC Technical Library (C422).
2. Pronounced "star property"
3. [redacted] was recently hired as Deputy Chief of C3.

P.L. 86-36

### Bibliography

1. Trusted Computer System Evaluation Criteria, CDC-STD-001-83, 15 Aug 83 (S-225,711).
2. ADP Security Certification/Accreditation Planning Guide, undated.
3. On the Feasibility of Connecting RECON to an External Network, [redacted] dated 16 Mar 81.

P.L. 86-36

# CRYPTOLOG 1983 INDEX (U)

*The following is a cumulative index of CRYPTOLOG (Vol X, 1983) and is in three parts, by title, by author, and by keyword. Items in multiple issues (January-February 1983, for example) are indicated by the first month (i.e., by Jan 83).*





## TITLES (U)

~~SECRET~~

P.L. 86-36

Letter to the Editor: Redbaron, Roadrunner..., Jun-Jul 83 Issue; Sep 83; [redacted]
Letter to the Editor: Security of Classified Information; Jun 83; [redacted]
Letter to the Editor: The Tower of Babel; May 83; Mollick J.J.
Letter to the Editor: Tips on Topical Reporting, Oct 83 Issue; Dec 83; [redacted]
Letter to the Editor: Tips on Topical Reporting, reply to [redacted] letter; Dec 83; [redacted]
Letter to the Editor: UNIX ED (I) Manual Page Comment; Oct 83; [redacted]
Letter to the Editor: Video Teleconferencing, Mar 83 Issue; Jun 83; [redacted]
The Literary Bends; Nov 83; Murphy A.I.
Logic Design Exceeding Boolean Capabilities; Jan 83; [redacted]
MBTI: The Management Tool of the Future; Nov 83; [redacted]
Man Does Not Live By Matzos Alone; Apr 83; 'Marian D. Librarian'
Management of Coordination; Sep 83; [redacted]
Managing Our Systems for Performance; Jan 83; [redacted]
Menu Selection As A Tool for Human/Machine Interaction; Jan 83; [redacted]
[redacted]
Mar 83; [redacted]
[redacted]
Dec 83; [redacted]
More on Passwords; Mar 83; [redacted]
My Staff--It Comforts Me; Apr 83; 'Zebulon Zilch'
NSA in The Space Age; Apr 83; [redacted]
The NSA High-level Display File; Jan 83; [redacted]
[redacted] Jun 83;
[redacted]
NSA-Crostic No. 46; Apr 83; Williams D.H.
NSA-Crostic No. 47; May 83; Williams D.H.
NSA-Crostic No. 48; Jun 83; Filby V.R.
NSA-Crostic No. 49; Aug 83; Williams D.H.
NSA-Crostic No. 50; Sep 83; Williams D.H.
NSA-Crostic No. 51; Dec 83; Williams D.H.
1982 Local Area Network Status; Jan 83; [redacted] E.M.
Non Posse vs. Posse Non; Dec 83; [redacted] H.G.
On How The 'Game' of the Agency Should Be Played; Sep 83; Santiago-Ortiz R.
Out of My Depth; May 83; P.L. 86-36
Out of My Depth; Dec 83;
PARPRO: Reconnaissance Programs; Sep 83; [redacted]
Picture: What Is The Caption?; Nov 83;
Punching The Biological Timeclock; Jun 83; [redacted]
Puzzle; Jan 83; Williams D.H.
Redbaron, Roadrunner, Bronzstar: What's In A Name?; Jun 83; [redacted]
Review: The Battle For The Falklands; Aug 83; [redacted]
Review: Digital Telephony; May 83; [redacted] J.A.

P.L. 86-36

SIGINT Challenge: A Scenario; Mar 83; [redacted] J.L.
Shell Game: System Shells; Dec 83; [redacted] W.E.
Shell Using If; Mar 83; [redacted]
Some Tips on Getting Promoted; Jun 83; [redacted] V.
Soviet Military Goals And Their Effect on Negotiations for Arms Limitations; Oct 83; [redacted]
Soviet Psi Experiments; Dec 83; [redacted]
Specifying Colors for Computer Graphics; Apr 83; [redacted]
Static Magic: The Wonderful World of Tempest; Nov 83; Donahue T.M.
Still More About Passwords; May 83; [redacted] M.E.
A Survey of Parallel Sorting; Jan 83; [redacted] S.B.
TDY Travail; Mar 83; Filby V.R.
TELECOM 83:; Oct 83; [redacted]
Tempest for Every Office; Nov 83; [redacted]
Thousands Miss Demonstration; Oct 83; [redacted] R.L.
Tips on Topical Reporting; Oct 83; [redacted]
A Tutorial on Color Theory and Human Color Perception for the Color Graphics Programmer; Jan 83; [redacted]
UIS: User Interface System Part One: Concept; Apr 83; [redacted]
UIS: User Interface System Part Two: Architecture; Apr 83; [redacted]
Video Teleconferencing: NSA Applications; Mar 83; [redacted]
Weather: A Key Intelligence Indicator; Mar 83; [redacted]
The White House Is Singing Our Song; Nov 83; Murphy A.I.
Why Pascal? (Why Not?); Jun 83; [redacted]
Word People at NSA; Apr 83; 'Dickson Airy'
Wrangler...One Tough Customer; Sep 83; [redacted]

P.L. 86-36

EO 1.4.(c)
P.L. 86-36

# AUTHORS (U)

---

Mar 83 Announcement: KRYPTOS Society Spring Meeting
Mar 83 Announcement: Students! (NCEUR Independent Study Programs)
Mar 83 Cumulative Index (1974-1982), Part 1: Authors
Mar 83 E.T. At NSA
Apr 83 Cumulative Index (1974-1982), Part 2: Titles
Apr 83 FBIS Latin American Reference Aid
May 83 Cumulative Index (1974-1982), Part 3: Keywords
May 83 Out of My Depth
Jun 83 Announcement: Two New Language Aids [redacted] and Chinese-English)
Aug 83 Announcement: Request for Copies of Jan-Feb 83 Issue (CISI Essay Contest)
Aug 83 Correction: Do You Know the Differences?, Jun-Jul 83 Issue
Sep 83 Announcement: Contributions Solicited for CRYPTOLOG Articles
Nov 83 Picture: What Is The Caption?
Dec 83 Correction: October 1983 CRYPTOLOG Issue Add to Classification: 'REL UK CAN AUS NZ'
Dec 83 Out of My Depth

P.L. 86-36

EO 1.4.(c)
P.L. 86-36

'Dickson Airy'
Apr 83 Word People at NSA

'Juan Tuthri'
Oct 83 Letter to the Editor: Management of Coordination, Sep 83 Issue

'Marian D. Librarian'
Apr 83 Frontier Dentist
Apr 83 Man Does Not Live By Matzos Alone

"H.G.R"
Nov 83 I Remember JFK

'Watt Zizname'
Nov 83 5-4-3 Puzzle

'Zebulon Zilch'
Apr 83 My Staff--It Comforts Me

[redacted]
Mar 83 [redacted]

[redacted]
Dec 83 DCL; The Direct Communications Link

[redacted]
Jun 83 Letter to the Editor: Video Teleconferencing, Mar 83 Issue

[redacted]
Aug 83 The Case of the 'Fowled-Up' CRITIC

[redacted]
Oct 83 Tips on Topical Reporting
Dec 83 Letter to the Editor: Tips on Topical Reporting, reply to Day's letter

[redacted]
Jan 83 Ada: Conquering the Tower of Babel

[redacted]
Mar 83 Cryptic Crossword #3

[redacted]
Jun 83 Punching The Biological Timeclock

[redacted]
Mar 83 More on Passwords
Apr 83 Specifying Colors for Computer Graphics
May 83 Still More About Passwords
Aug 83 Does Your Office Make You Sick?
Nov 83 MBTI: The Management Tool of the Future

[redacted]
Jun 83 Punching The Biological Timeclock

[redacted]
Mar 83 [redacted]

[redacted]
Dec 83 Letter to the Editor: Tips on Topical Reporting, Oct 83 Issue

[redacted]
Sep 83 PARPRO: Reconnaissance Programs

[redacted]
Nov 83 Static Magic: The Wonderful World of Tempest

[redacted]
Mar 83 Weather: A Key Intelligence Indicator

[redacted]
Nov 83 [redacted]

P.L. 86-36

EO 1.4.(c)
P.L. 86-36

P.L. 86-36

[redacted]
Oct 83 Thousands Miss Demonstration

Faurer L.D.
Oct 83 Crisis Management: Remarks

[redacted]
Jan 83 Ada News

Filby V.R.
Mar 83 TDY Travail
Jun 83 NSA-Crostic No. 48

[redacted]
Mar 83 SIGINT Challenge: A Scenario

[redacted]
Jan 83 Improving Raster Graphics Images by Anti-Aliasing

[redacted] EO 1.4.(c)
Jan 83 Getting Personal P.L. 86-36

[redacted]
Aug 83 Letter to the Editor: Government of the People..., Apr 83 Issue

[redacted]
Sep 83 Management of Coordination
Nov 83 Acronymania

[redacted] P.L. 86-36
May 83 Confessions of a Briefer

[redacted]
Apr 83 UIS: User Interface System Part One: Concept

[redacted]
Jun 83 [redacted]

[redacted]
Apr 83 [redacted]

P.L. 86-36

[redacted]
Jun 83 Some Tips on Getting Promoted

EO 1.4.(d)

[redacted]
Nov 83 Letter to the Editor: Computerizing of TA, May 83 Issue

[redacted]
Mar 83 Shell Using If

[redacted]
Aug 83 Letter to the Editor: Out of My Depth, May 83 Issue

[redacted]
Jun 83 Why Pascal? (Why Not?)

[redacted]
Oct 83 Soviet Military Goals And Their Effect on Negotiations for Arms Limitations

P.L. 86-36

P.L. 86-36

[redacted]
Dec 83 The Islamic Time Bomb

[redacted]
Oct 83 Letter to the Editor: UNIX ED (I) Manual Page Comment

[redacted]
Apr 83 NSA in The Space Age

[redacted]
Nov 83 Tempest for Every Office

[redacted]
Jun 83 Redbaron, Roadrunner, Bronzstar: What's In A Name?

[redacted]
Apr 83 Government of the People, By The Party, For The Leadership
Aug 83 Letter to the Editor: Government of the People... reply to [redacted] letter

[redacted]
Apr 83 [redacted]
May 83 Cryptography At GLOBECOM 82
May 83 Review: Digital Telephony
Aug 83 Review: The Battle For The Falklands
Sep 83 Foreign Microwave Radio EO 1.4.(c)
Oct 83 TELECOM 83: P.L. 86-36
Dec 83 Soviet Psi Experiments

[redacted]
May 83 Letter to the Editor: The Tower of Babel

[redacted]
Jan 83 A Survey of Parallel Sorting

[redacted]
Sep 83 Do You Really Mean Julian?

Murphy A.I.
Nov 83 The Literary Bends
Nov 83 The White House Is Singing Our Song

[redacted]
Jan 83 Computer Graphics to Enhance Collection Management

[redacted]
Jan 83 The Future Brightens for Flat-Panel Displays

[redacted]
Jan 83 1982 Local Area Network Status

[redacted]
Jun 83 Do You Know the Differences?

[redacted]
Mar 83 Is The Glass Half Empty Or Half Full?

P.L. 86-36

P.L. 86-36

[redacted]

Jun 83 [redacted]

Aug 83 I Remember Mabel Babel
Aug 83 Letter to the Editor: My Staff--It Comforts Me, Apr 83 Issue
Dec 83 Non Posse vs. Posse Non

[redacted]

Jan 83 Menu Selection As A Tool for Human/Machine Interaction

[redacted]

Sep 83 Letter to the Editor: Redbaron, Roadrunner..., Jun-Jul 83 Issue

[redacted]

Sep 83 On How The 'Game' of the Agency Should Be Played

[redacted]

Jan 83 The NSA High-level Display File

[redacted]

Apr 83 UIS: User Interface System Part Two: Architecture

[redacted]

Jan 83 Logic Design Exceeding Boolean Capabilities

[redacted]

Dec 83 [redacted]

[redacted]

Mar 83 Video Teleconferencing: NSA Applications

[redacted]

May 83 The Intelligence Watch Officer

[redacted]

Jun 83 [redacted]

[redacted]

Oct 83 Banners, Cowboy Hats, and ELINT Notations

[redacted]

May 83 Computerizing Traffic Analysis
Dec 83 Shell Game: System Shells

[redacted]

Sep 83 [redacted]
Sep 83

[redacted]

Jan 83 Managing Our Systems for Performance

[redacted]

Jan 83 A Tutorial on Color Theory and Human Color Perception for the Color Graphics Programmer

P.L. 86-36

P.L. 86-36

[redacted]

Sep 83 Wrangler...One Tough Customer

[redacted]

Jun 83 Letter to the Editor: Security of Classified Information

Williams D.H.
Jan 83 Puzzle
Apr 83 NSA-Crostic No. 46
May 83 NSA-Crostic No. 47
Aug 83 NSA-Crostic No. 49
Sep 83 NSA-Crostic No. 50
Dec 83 NSA-Crostic No. 51

[redacted]

Jan 83 Computer Graphics to Enhance Collection Management

EO 1.4.(c)
P.L. 86-36


P.L. 86-36

EO 1.4.(c)
P.L. 86-36

P.L. 86-36

# KEYWORDS (U)

P.L. 86-36



P.L. 86-36

HUMINT
Jun 83 [redacted]

Humor
Apr 83 Frontier Dentist; 'Marian D. Librarian'
Apr 83 Man Does Not Live By Matzos Alone; 'Marian D. Librarian'
Apr 83 My Staff--It Comforts Me; 'Zebulon Zilch'
Apr 83 NSA in The Space Age; [redacted]
Apr 83 Word People at NSA; 'Dickson Airy'
May 83 Letter to the Editor: The Tower of Babel; [redacted]
Aug 83 Letter to the Editor: My Staff--It Comforts Me, Apr 83 Issue; [redacted]

Index
Mar 83 Cumulative Index (1974-1982), Part 1: Authors;
Apr 83 Cumulative Index (1974-1982), Part 2: Titles;
May 83 Cumulative Index (1974-1982), Part 3: Keywords;

Indicators
Mar 83 Weather: A Key Intelligence Indicator; [redacted]

Iran
Dec 83 The Islamic Time Bomb; [redacted] F.W.

P.L. 86-36

Islam
Dec 83 The Islamic Time Bomb; [redacted] F.W.

[redacted]
Aug 83 The Case of the 'Fowled-Up' CRITIC; [redacted]

P.L. 86-36

KRYPTOS News
Mar 83 Announcement: KRYPTOS Society Spring Meeting;

Language
Jun 83 Announcement: Two New Language Aids [redacted] and Chinese-English);
Jun 83 [redacted]
Aug 83 I Remember Mabel Babel [redacted]

Latin American
Apr 83 FBIS Latin American Reference Aid;

Linguists
Dec 83 Non Posse vs. Posse Non; [redacted] H.G.

Logic
Jan 83 Logic Design Exceeding Boolean Capabilities; [redacted]

EO 1.4.(c)
P.L. 86-36

EO 1.4.(c)
P.L. 86-36

P.L. 86-36

Management
Sep 83 On How The 'Game' of the Agency Should Be Played; Santiago-Ortiz R.
Oct 83 Crisis Management: Remarks; Faurer L.D.
Nov 83 MBTI: The Management Tool of the Future; [redacted]

Mathematics
Jan 83 Logic Design Exceeding Boolean Capabilities; [redacted]

Microcomputers
Jan 83 Getting Personal; [redacted]

Microwave
Sep 83 Foreign Microwave Radio; [redacted]

Mx
Jan 83 Logic Design Exceeding Boolean Capabilities; [redacted]

EO 1.4.(c)
P.L. 86-36

NCS
Mar 83 Announcement: Students! (NCEUR Independent Study Programs);
Nov 83 The Literary Bends; Murphy A.I.
Nov 83 The White House Is Singing Our Song; Murphy A.I.

NSOC
May 83 The Intelligence Watch Officer; [redacted] L.S.

[redacted]
Mar 83 [redacted]

Pascal
Jun 83 Why Pascal? (Why Not?); [redacted]

Password
Mar 83 More on Passwords; [redacted]
May 83 Still More About Passwords; [redacted] M.E.

Performance
Jun 83 Punching The Biological Timeclock; [redacted]
Aug 83 Does Your Office Make You Sick?; [redacted]

Personality
Mar 83 E.T. At NSA;
Apr 83 Word People at NSA; 'Dickson Airy'
Aug 83 I Remember Mabel Babel; [redacted]
Nov 83 I Remember JFK; H.G.R

Promotions
Jun 83 Some Tips on Getting Promoted; [redacted] V.

P.L. 86-36

EO 1.4.(c)
P.L. 86-36

~~SECRET~~

P.L. 86-36

Puzzle
Jan 83 Puzzle; Williams D.H.
Mar 83 Cryptic Crossword #3; [redacted]
Apr 83 NSA-Crostic No. 46; Williams D.H.
May 83 NSA-Crostic No. 47; Williams D.H.
May 83 Out of My Depth;
Jun 83 NSA-Crostic No. 48; Filby V.R.
Aug 83 Letter to the Editor: Out of My Depth, May 83 Issue; [redacted]
Aug 83 NSA-Crostic No. 49; Williams D.H.
Sep 83 NSA-Crostic No. 50; Williams D.H.
Nov 83 5-4-3 Puzzle; 'Watt Zizname'
Dec 83 NSA-Crostic No. 51; Williams D.H.
Dec 83 Out of My Depth;

Reconnaissance
Sep 83 PARPRO: Reconnaissance Programs; [redacted]

Reporting
Aug 83 The Case of the 'Fowled-Up' CRITIC; [redacted]
Oct 83 Tips on Topical Reporting; [redacted]
Dec 83 Letter to the Editor: Tips on Topical Reporting, Oct 83 Issue; [redacted]
Dec 83 Letter to the Editor: Tips on Topical Reporting, reply to Day's letter; [redacted] D.G.

EO 1.4.(c)
P.L. 86-36

Satellites
Apr 83 [redacted]

Security
Jun 83 Do You Know the Differences?; [redacted] D.S.
Jun 83 Letter to the Editor: Security of Classified Information; [redacted]
Aug 83 Correction: Do You Know the Differences?, Jun-Jul 83 Issue;

SIGINT
Mar 83 SIGINT Challenge: A Scenario; [redacted] J.L.

Sorting
Jan 83 A Survey of Parallel Sorting; [redacted] S.B.

P.L. 86-36

Soviet
Apr 83 Government of the People, By The Party, For The Leadership; [redacted]
Aug 83 Letter to the Editor: Government of the People... reply to [redacted] letter; [redacted]
Aug 83 Letter to the Editor: Government of the People..., Apr 83 Issue; [redacted]
Oct 83 Soviet Military Goals And Their Effect on Negotiations for Arms Limitations; [redacted] G.L.
Dec 83 Soviet Psi Experiments; [redacted]

Spanish
Jun 83 [redacted]

P.L. 86-36

EO 1.4.(c)
P.L. 86-36

Dec 83 Non Posse vs. Posse Non; [redacted] H.G.

Staff
Apr 83 My Staff--It Comforts Me; 'Zebulon Zilch'

TDY
Mar 83 TDY Travail; Filby V.R.

TELECOM
Oct 83 TELECOM 83:; [redacted]

Tempest
Nov 83 Static Magic: The Wonderful World of Tempest; Donahue T.M.
Nov 83 Tempest for Every Office; [redacted]

Terminology
Sep 83 Do You Really Mean Julian?; [redacted] C.

Time
Sep 83 Do You Really Mean Julian?; [redacted] C.

Traffic Analysis
May 83 Computerizing Traffic Analysis; [redacted]
Nov 83 Letter to the Editor: Computerizing of TA, May 83 Issue; [redacted]

Training
Mar 83 Announcement: Students! (NCEUR Independent Study Programs);

UIS
Apr 83 UIS: User Interface System Part One: Concept; [redacted]
Apr 83 UIS: User Interface System Part Two: Architecture; [redacted]

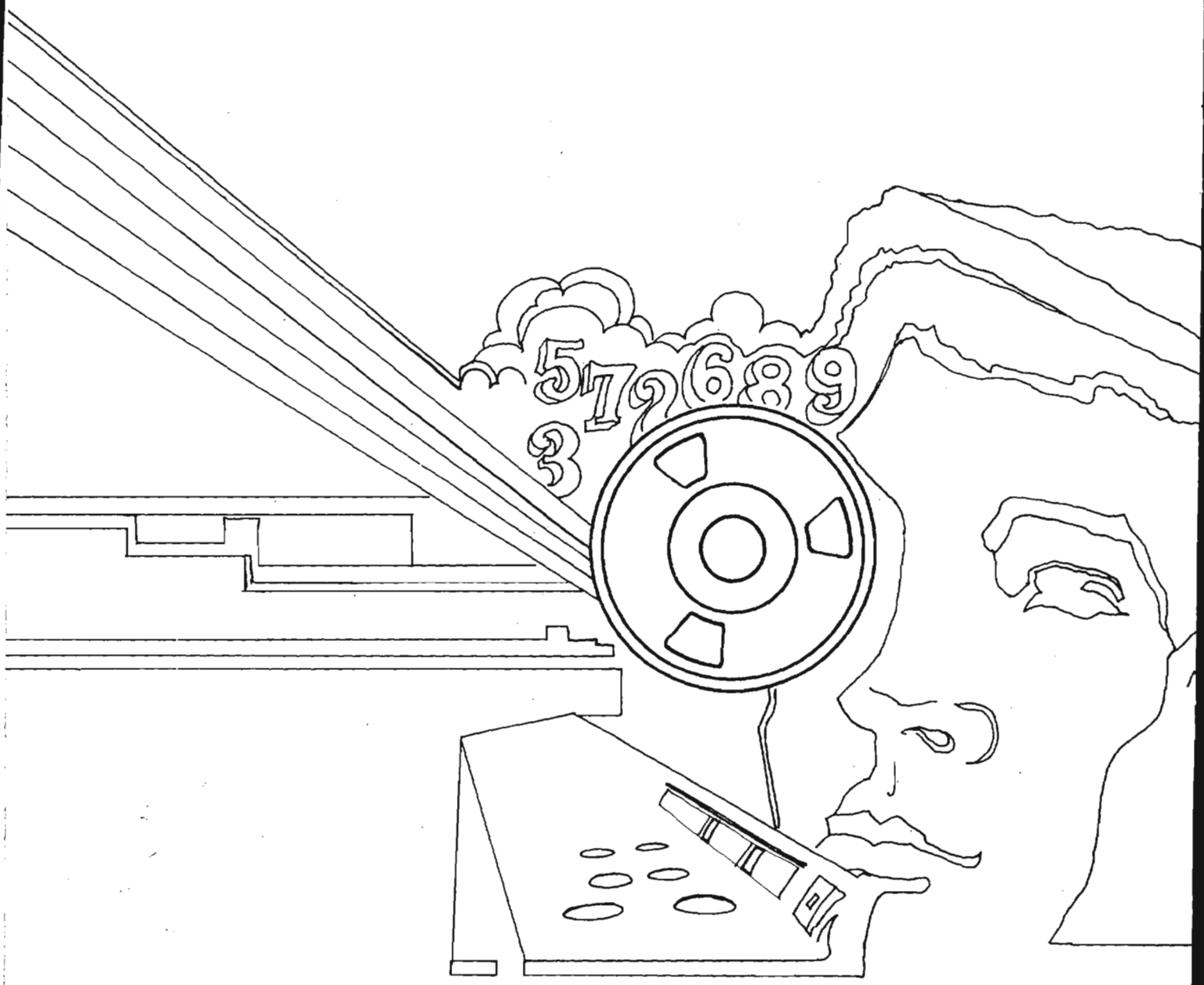
UNIX
Mar 83 Shell Using If; [redacted]
Oct 83 Letter to the Editor: UNIX ED (I) Manual Page Comment; [redacted]
Dec 83 Shell Game: System Shells; [redacted] W.E.

Video Teleconferencing
Mar 83 Video Teleconferencing: NSA Applications; Snodgrass C.L.
Jun 83 Letter to the Editor: Video Teleconferencing, Mar 83 Issue; [redacted] J.R.

Weather
Mar 83 Weather: A Key Intelligence Indicator; [redacted]

P.L. 86-36

Writing
Nov 83 The Literary Bends; Murphy A.I.
Nov 83 The White House Is Singing Our Song; Murphy A.I.




~~HANDLE VIA COMINT CHANNELS ONLY~~