Global Information Assurance Certification Paper

CA-ACF2 USER ACCOUNT CLEANUP

Scott C. Meyer
16 February 2004
GSEC Practical V1.4b - Option 2

ABSTRACT

This paper discusses the cleanup of a user base on a CA-ACF2 system. The case study involves a large company whose system administrators have been historically negligent on identifying employees leaving the company and removing their access. I will identify key privileges and access definitions within the logonid records that can be searched to identify the accounts that may require attention. Once identified, appropriate steps to remove the accounts will be described.

While explaining the case study, this paper will introduce some key aspects of CA-ACF2 and the properties of user accounts. I will also describe ACF commands used to modify and delete logonid records within the system. After completion of the cleanup, the environment will have a reduced risk of unauthorized use, more efficient use of system resources, and processes implemented to prevent future accumulation of unauthorized accounts.

BEFORE

This case study takes place at a major medical insurance company that I will refer to as XYZ Insurance. XYZ Insurance implemented CA-ACF2 in the early 1980's on OS/390 as their primary business tool for processing claims, and users accessed this data via dumb terminals connected to the mainframe. The large user base was easy to administrate by a small number of account administrators due to the fact that it was the only data system in the company.

During the mid to late 1990's, XYZ Insurance chose to upgrade their systems to a Microsoft Windows environment. Over the next 10 years, many new applications and networks were installed such as Novell, Windows NT domain, UNIX, Lotus Notes, LDAP, Sybase, Oracle, etc. To ease the process of requesting access to any of the many systems available, a single department for user account administration (UAA) was formed. Procedures and budget focused on staffing the department sufficiently for user account creation and maintenance. However, little resources were available for removing accounts of former employees or other unused accounts for risk management.

XYZ Insurance invested in an audit of their systems by an external entity in order to obtain an objective evaluation of their systems status for HIPAA security and

privacy compliance. Many areas were discovered that required attention, but XYZ Insurance did have ample time to correct the issues before the scheduled compliance deadlines as described at http://www.hipaacomply.com/timeline.htm. The audit of the CA-ACF2 system revealed a large number of both active and inactive user accounts that did not correlate with the current actual users of the system. This presented a security risk of unauthorized and un-auditable access to the system. A cleanup effort had to be implemented to reduce the accounts to those only used by current employees and/or system processes.

DURING

A) Plug the Hole

The first step taken for the cleanup effort was to develop policy and procedures to prevent the accumulation of more unused accounts. The original policy at XYZ Insurance was for the supervisors of ex-employees to submit requests to UAA to remove access, which had been in place since the deployment of CA-ACF2. Given the size of the company and the number of supervisors, this left many points of failure in the process, and UAA was not receiving appropriate notice to remove user access.

It would be best to receive notification of employees leaving the company from a single source. The logical answer for this single source was the Human Resources department (HR). New policies and procedures were implemented allowing UAA to have access to the database of employees. This allowed UAA to quickly run daily reports of newly removed employees, as well as new hires and departmental transfers. Now UAA could actively revoke access immediately to former employees and prepare their accounts for deletion. The SUSPEND bit on the logonid record is set per the new procedure to revoke access to the account. These new procedures prevented the accumulation of unused and unauthorized accounts on all systems, including CA-ACF2.

B) Identify and Delete Inactive Accounts

As with most all systems, user accounts in CA-ACF2 are either active or inactive. Since it is much easier to identify inactive account than singling out active accounts that are not associated with a current employee, it was decided to cleanup the inactive account first.

CA-ACF2 accounts can be inactivated in three fashions. The following is an example of a logonid record, 123456, that is inactivated by use of all three locking fields. Within the record example, I have highlighted and boldfaced four potions of the logonid that will play crucial roles in the search for all unused accounts. These fields are CANCEL, SUSPEND, EXPIRE(*date*), and ACC-DATE(*date*).

```
123456          NYCCLMPRCE        123456 BEANSTALK, JACK B
                DEPT(CLM) JOBF(PRC) LOC(NYC) STS(E)
CANCEL/SUSPEND  CANCEL CSDATE(11/11/03) CSWHO(112233) SUSPEND
PRIVILEGES      ACTIVE(10/08/01) CICS EXPIRE(11/12/03) JOB TSO
ACCESS          ACC-CNT(61) ACC-DATE(10/09/03) ACC-SRCE(TERM3879)
                ACC-TIME(10:08)
PASSWORD        MAXDAYS(33) MINDAYS(3) PSWD-DAT(00/00/00) PSWD-INV(0)
                PSWD-SRC(TERM3819) PSWD-TIM(10:24)
                PSWD-TOD(11/05/03-10:55) PSWD-VIO(0)
TSO             DFT-PFX(USR) INTERCOM JCL LGN-SIZE MAIL NOTICES
                OPERATOR PROMPT TSOPROC($TSO) TSORBA(0000CB)
                TSORGN(4,000) TSOSIZE(9,000) TSOTIME(1,440) WTP
STATISTICS      SEC-VIO(0) UPD-TOD(11/09/03-10:08)
CICS            CICSID(U5J) CICSPRI(4)
RESTRICTIONS    GROUP(A0070000) PREFIX(123456)
USER FIELDS     MEMO(CLAIMS PROCESSOR - NYC       )
```

The following are the definitions of these fields as defined in Chapter 3 of CA-ACF2 Administrator Guide:

> **ACC-DATE(*date*)**: The date of the last system access by this user.
> **CANCEL**: Specifies that the logonid cannot be used to access the system.
> **EXPIRE(*date*)**: Indicates when the privileges for this logonid will expire. When the specified date is reached, the user is no longer able to log on or submit jobs.
> **SUSPEND**: Specifies that the logonid cannot be used to access the system.

CANCEL and SUSPEND are interchangeable in functionality and are not differentiated by CA-ACF2. The date fields are specified in the format of mm/dd/yy, dd/mm/yy, or yy/mm/dd, depending on selection of the DATE field of the GSO OPTS record. In my experience though, the most commonly used format is mm/dd/yy, and that is the format used in this case study. In preparation of 'Year 2000' issues with these 2-numeral date fields, CA-ACF2 is set so that 70-99 indicates the years of 1970-1999, and 00-69 indicates the years of 2000-2069.

1) Identify EXPIRE(*date*) Accounts

The first focus of identifying accounts to cleanup were accounts with the EXPIRE field set to any past date. The ACC-DATE field was also incorporated into the search so as to find all EXPIRED accounts that had not been accessed in the last 30 days. This "30 day" selection was used to prevent the deletion of any account that may have been expired inappropriately. 30 days was deemed a safe buffer.

Note: For better understanding of the dates in all the commands used in this case study, please assume that the commands were performed on December 1, 2003.

From the ACF command prompt, the following commands was executed:

```
SET LID
SET TERSE
LIST IF(EXPIRE >= D'01/01/70' AND EXPIRE < D'12/01/03' -
AND ACC-DATE < D'11/01/03')
```

The first command verifies that the system is to process logonid records within ACF. The second command sets the system to display the short version of the logonid record. The short version is essentially only the first line of the example record on previous page. Therefore, the results displayed of the third command will only include the user's unique ID and the NAME field.

The third command asks the system to list all logonid records that have an EXPIRE date from January 1, 1970 to November 30, 2003 and that have not been accessed prior to November 1, 2003. It may be questioned on why a date range has been specified for the EXPIRE date rather than request all accounts with an EXPIRE date less than December 1, 2003. The reason is in the logic of how CA-ACF2 interprets accounts without an EXPIRE date. A logonid record always has an EXPIRE date set. For account where the EXPIRE date doesn't function, the value of the date is 00/00/00. Therefore, a search of all dates less than 12/01/03 would have also produced results where all account have EXPIRE set to 00/00/00. These needed to be excluded.

The list was copied into an Excel spreadsheet for documentation, review, and a proper audit trail.

2) Delete Identified Expired Accounts

Now that the list of accounts meeting the specified parameters was available, it was first reviewed manually against a list from HR of current employees to identify any account expired due to extended leave of absence or extended vacation. It was also reviewed for instances of any specific system accounts used intermittently. An example of such an account would be a firecall id or an account that ran periodic batch jobs. The company did not want to lose any productivity due to the incorrect deletion of any account.

Once the accounts were confirmed for deletion, the following ACF command was executed:

```
DEL 123456
```

'123456' was the logonid for the use in the example logonid record on page 3. The command would be repeated for every unique logonid on the report.

3) Identify Cancelled and Suspended Accounts
After the revoked accounts with EXPIRE dates set were removed from the system, the accounts with the fields CANCEL and SUSPEND turned on were handled next. Since both fields have the same affect in CA-ACF2, it was easy to handle both settings in the same step. Once again, concerns about removing accounts incorrectly marked as CANCEL or SUSPEND was handled by reviewing those accounts that had not been accessed for 30 days or more as a safe buffer.

From the ACF command prompt, the following commands were executed:

```
SET LID
SET TERSE
LIST IF(SUSPEND AND ACC-DATE < D'11/01/03')
```

And

```
SET LID
SET TERSE
LIST IF(CANCEL AND ACC-DATE < D'11/01/03')
```

This produced a list of all accounts that were suspended or cancelled and had last been accessed prior to November 1, 2003. These 2 lists were also copied into an Excel spreadsheet for documentation and review. This list was also reviewed for duplication of account listings. Since it is possible for both the CANCEL and SUSPEND bits to be turned on at the same time, duplication on the final list was possible. Duplicates were removed for simplification of review and deletions.

4) Delete Identified Cancelled and Suspended Accounts
This execution of this step is identical to the process I described in Step #2 on page 4.

5) Identify Excluded Accounts Due to 30 Day Buffer
The list commands from steps 1 and 3 above excluded accounts that had been accessed during the prior 30 days. The easiest way to identify these accounts and maintain the 30 day buffer as requested by the company was to simply wait 30 days and execute slightly modified LIST commands again.

On January 2, 2004 the following commands were executed:

```
LIST IF(EXPIRE >= D'01/01/70' AND EXPIRE < D'12/01/03' -
AND ACC-DATE >= D'11/01/03' AND ACC-DATE <= D'12/01/03')

LIST IF(SUSPEND AND ACC-DATE >= D'11/01/03' AND -
ACC-DATE <= D'12/01/03')
```

```
LIST IF(CANCEL AND ACC-DATE >= D'11/01/03' AND -
ACC-DATE <= D'12/01/03')
```

These three list commands supplied the user accounts that were revoked and had not accessed the system between November 1, 2003 and December 1, 2003. It was not necessary to review what the exact date of access was since I had already waited 30 days before compiling the report. Once again, before deleting these accounts, the list was reviewed and documented in the same fashion listed in step 2.

C) Identify and Delete Unused Active Accounts

Now that all revoked accounts have been identified and appropriately deleted, the remaining accounts to address for cleanup are those that have not been used for access for an extended period of time. There were still a large number of accounts remaining on the system, many more than the number of current employees. The trick was to find an efficient, yet accurate way to determine the validity of each account. The answer came from XYZ Insurance's interpretation of HIPAA regulations for minimum required user access and role-based access. (http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=2003_register&docid=fr20fe03-4.pdf) XYZ Insurance made the decision to migrate to a role-based access strategy for all employees as their new system access policy.

Given this interpretation, XYZ Insurance agreed that any account that had not been accessed for greater than 90 days was deemed as not necessary for the user's daily job function. Multiple communications were sent to employees via email apprising them of this new policy and to forewarn them of possible CA-ACF2 account deletion if it had not been used within 90 days. The communication further explained that should a user need access after an account was deleted, an appropriate access request would need to be submitted to the UAA department.

The following commands were executed from the ACF prompt:

```
SET LID
SET TERSE
LIST IF(NOCANCEL AND NOSUSPEND AND EXPIRE = D'00/00/00' -
AND ACC-DATE < D'09/01/03')
```

Since the accounts with CANCEL, SUSPEND, and EXPIRE fields have already been addressed, these accounts were excluded from this listing. This left a list of accounts that had not been accessed for more than 90 days. The list was copied into an Excel spreadsheet once again for documentation and review. The list was reviewed for any instances of system accounts or firecall accounts deemed necessary to maintain the system. These accounts were removed from the list and all remaining account deleted. Procedures were also adopted so that this

listing would be run monthly in order to remove these types of unused yet active accounts.

D) Review of Remaining Accounts
At this point, thousands of accounts have been removed from the system. The remaining accounts should be identifiable with either a current user or a current documented system process. However, this assumption cannot be believed as true. All accounts remaining on the system should be reviewed and compared against a list of current employees from HR and a list of system processes. A full list of the remaining CA-ACF2 accounts can be produced by executing the following ACF commands:

```
SET LID
SET TERSE
LIST UID()
```

Once the list was compiled, it was reviewed. Accounts that could not be linked to current employees or processes were not immediately deleted. Instead, they were first set to CANCEL for a period of 30 days. If the accounts were indeed needed for authorized business needs, it would soon become apparent to the user of the account that it was revoked and would seek assistance to restore its access. At this point, the account was adequately documented and restored for use. After 30 days, the remaining unauthorized accounts identified in this step were deleted.

E) Review Dataset and Resource Rule
By design, access to any dataset or resource in CA-ACF2 is denied by default. Specific access rules must be written to govern the access to them. Typical deployment strategies or CA-ACF2 in a business incorporates the use of the UID string to develop role-based access. From the example logonid record on page, we can see that the full UID string for Jack Beanstalk is:

```
(NYCCLMPRCE       123456)
```

The portion of the UID string listing '123456' is the unique logonid for Jack. For XYZ Insurance, this unique logonid happens to be the same as the unique employee ID issued to him by HR. The first portion, 'NYCCLMPRCE' indicates flags on the UID string to be used to grant default access. The key flags that it indicated are as follows:

LOC(NYC) – This states his office *location* is in New York City
DEPT(CLM) – This states that he is a member of the claims *department*.
JOBF(PRC) – This states that his *job function* is a claims processor.
STS(E) – This states that Jack's *status* is a full time employee at XYZ Insurance.

Rules to datasets and resources can be written to grant default access to users with any one of or combination of these 4 flags. This promotes the role-based

access and reduces system administrative work. However, there are common exceptions to the role-based access where specific access needs to be granted to a specific logonid. These rules for individual access need to be cleaned up after the deletion of a logonid record.

In the CA-ACF2 environment developed at XYZ Insurance, it would not be as critical to remove these rules since logonids will always be unique given the use of a unique employee number issued by HR. However, many CA-ACF2 environments are administrated to have the unique logonid be a formula of the user's name such as first initial, middle initial, then first 5 letters of the last name. This sort of setup would have given our example account for Jack Beanstalk the unique logonid of JBBEANS. With the logonid record for Jack deleted, access to any areas specified for JBBEANS would still be inaccessible unless a person was later hired by the name Jennifer B Beansprout. By definition, her logonid would also be JBBEANS, thus inadvertently granting her access to areas intended initially only for Jack Beanstalk. This demonstrates the importance to cleanup the access rules during this account cleanup project.

Typically, the access rules review is the most time consuming. At XYZ Insurance, each access rule was studied for any listings of access granted to a specific logonid. If any were found, those logonids were compared to a list of all current logonid records. If not a current logonid, the access definition was removed from the rule. Then the rule was compiled. It was important to use a list of current logonids for the comparison rather than the list of logonids that were deleted in our cleanup process. This prevented the retention of logonids within rules where the logonid record had been deleted prior to the cleanup effort but not removed from access rules.

As a safeguard, a temporary procedure should also be implemented to the account creation process while the project of reviewing all access rules is in progress. Since it is difficult to quickly identify the uniqueness of a new logonid compared to any listing within all the access rule sets, it is important to keep a central log file of all newly created logonids and any specific access rules written for them. This log should then be included in the research of logonids found in the project of reviewing access rules.

F) Review Administrative Privileges

This was the final step in the project. With the development of new procedures and policies discussed in the previous sections, it was important to insure that all future account creations, account deletions, account modifications, and access rule maintenance complied with them. It was decided that only appropriately trained personnel in the User Access Administration department should have these rights. A list of all personnel with rights to perform these functions was needed for review.

The ability to create, delete, and modify logonids and create or modify access rules is determined by three privilege bits on the logonid record. They are SECURITY, ACCOUNT, and LEADER. Definitions for these privileges are:

SECURITY: Users with this privilege are known as security administrators. These users have access to view and modify all logonid records, datasets, resources, and access rules. They can also create new access rules and delete current rules. However, this privilege does not allow the creation or deletion of a logonid record.
ACCOUNT: Users with this privilege have rights to create, delete, view, and modify any logonid record. The only limitation to the logonid record is that it will not allow the addition of the SECURITY privilege to an account.
LEADER: Users with this privilege have limited rights to modify selected fields within the logonid record. The select fields that they can modify are defined in the ACF Field Definition Record (ACFFDR) which defines many parameters of the CA-ACF2 environment.

Users that have these privileges can be listed with the following ACF commands:

```
LIST IF(SECURITY)
LIST IF(ACCOUNT)
LIST IF(LEADER)
```

Once these users are identified, it can be reviewed whether their current job functions require the administrative access. If so, these users need to be aware of all new policies and procedures to insure a low security risk to the system. If the user does not require the access, it can be removed with one of the following ACF commands:

```
CHANGE logonid NOSECURITY
CHANGE logonid NOACCOUNT
CHANGE logonid NOLEADER
```

AFTER

During this described cleanup process, thousands of accounts were deleted from the CA-ACF2 system. The remaining accounts had all been verified as necessary for daily production needs of XYZ Insurance. There were some complications created in the process when a few users' logonid records were deleted but were still needed. These accounts represented less than 0.5% of the total user base however, and the issues were quickly resolved with the immediate recreation of the authorized accounts. Processes have been defined to delete all future former employees in a timely manner and to recognized and delete unused accounts to prevent the accumulation of extraneous accounts on the system.

A side benefit of this cleanup was the increase in performance and efficiency of the system. With the deletion of so many logonid records, database access rules, and resource access rules, the datasets that house this access information has significantly reduced in size. This allowed CA-ACF2 to increase its efficiency when verifying access requests.

The specific processes that I described in this case study for the cleanup project are not the only way to perform the work. These were simply the steps that are available on all CA-ACF2 setups. The cleanup process could be more efficient if other applications and/or interfaces are available like the ISPF Panels, Cleanup for ACF2 Security, CA-ACF2 Security Workstation, or ETF/A. (http://www.ekcinc.com/) (www3.ca.com/Solutions/ProductsAZ.aspx) These are available from either Computer Associates or EKC, Inc. but are also costly programs to acquire.

After the completion of the cleanup, XYZ Insurance invested in a second external audit for an objective review of compliance with HIPAA regulations. The audit was positive. The overall security risks to the CA-ACF2 system were minimal and all fell within HIPAA compliance.

BIBLIOGRAPHY

HIPAAcomply
URL: http://www.hipaacomply.com (8 January 2004)

HIPAAcomply Timeline
URL: http://www.hipaacomply.com/timeline.htm (8 January 2004)

HIPAA.ORG Security Final Rule
URL: http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=2003_register&docid=fr20fe03-4.pdf (20 February 2003)

EKC, Ind. ETF/A Product Resources.
URL: http://www.ekcinc.com/MSD-etfahomepage.htm (9 January 2004)

Computer Associates International, Inc. eTrust CA-ACF2 Security for z/os and OS/390, Administration Guide 6.5 Computer Associates International, Inc. 2002 Chapter 3. pp 44-65.

EKC, Inc ACF2: Fundamentals of Daily Administration Eberhard Klemens Co. Inc. 2003. Privileges 4 – 5.