



```

      AAAAAA    UU      UU    TTTTTTTTTT    HH      HH    000000    RRRRRRRR    IIIIII    ZZZZZZZZZZ    EEEEEEEEEE
      AAAAAA    UU      UU    TTTTTTTTTT    HH      HH    000000    RRRRRRRR    IIIIII    ZZZZZZZZZZ    EEEEEEEEEE
AA      AA    UU      UU    TT          HH      HH    00      00    RR      RR    II      II    ZZ          EE
AA      AA    UU      UU    TT          HH      HH    00      00    RR      RR    II      II    ZZ          EE
AA      AA    UU      UU    TT          HH      HH    00      00    RR      RR    II      II    ZZ          EE
AA      AA    UU      UU    TT          HH      HH    00      00    RR      RR    II      II    ZZ          EE
AA      AA    UU      UU    TT          HHHHHHHHHH    00      00    RRRRRRRR    II      II    ZZ          EEEEEEEE
AA      AA    UU      UU    TT          HHHHHHHHHH    00      00    RRRRRRRR    II      II    ZZ          EEEEEEEE
AAAAAAAAAA    UU      UU    TT          HH      HH    00      00    RR      RR    II      II    ZZ          EE
AAAAAAAAAA    UU      UU    TT          HH      HH    00      00    RR      RR    II      II    ZZ          EE
AA      AA    UU      UU    TT          HH      HH    00      00    RR      RR    II      II    ZZ          EE
AA      AA    UU      UU    TT          HH      HH    00      00    RR      RR    II      II    ZZ          EE
AA      AA    UUUUUUUUUU    TT          HH      HH    00      00    RR      RR    IIIIII    ZZZZZZZZZZ    EEEEEEEEEE
AA      AA    UUUUUUUUUU    TT          HH      HH    000000    RR      RR    IIIIII    ZZZZZZZZZZ    EEEEEEEEEE

```

```

      CCCCCCCC    LL      DDDDDDDD
      CCCCCCCC    LL      DDDDDDDD
CC      LL      DD      DD
CC      LL      DD      DD
CC      LL      DD      DD
CC      LL      DD      DD
CC      LL      DD      DD
CC      LL      DD      DD
CC      LL      DD      DD
CC      LL      DD      DD
CC      LL      DD      DD
CCCCCCCC    LLLLLLLLLL    DDDDDDDD
CCCCCCCC    LLLLLLLLLL    DDDDDDDD

```

DAF  
MOC  
/\*  
/\*  
/\*-  
ags

module authorize\_commands

Version: 'V04-000'

```

*****
*
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
* ALL RIGHTS RESERVED.
*
* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
* TRANSFERRED.
*
* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
* CORPORATION.
*
* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*
*****

```

MODIFIED BY:

- V03-009 JRL0027 John R. Lawson, Jr. 25-Jul-1984 10:54  
Add MODIFY/SYSTEM\_PASSWORD=xxxxx command to modify the system password.
- V03-008 JRL0015 John R. Lawson, Jr. 27-Jun-1984 10:03  
Report UAF\$\_ZISQUAL if user forgets qualifier on GRANT, etc.
- V03-007 JRL0012 John R. Lawson, Jr. 21-Jun-1984 15:58  
Add /GENERATE\_PASSWORD qualifier to generate random passwords.
- V03-006 JRL0008 John R. Lawson, Jr. 20-Jun-1984 14:02  
Add /PWDEXPIRED qualifier to allow pre-expired passwords.
- V03-005 JRL0001 John R. Lawson, Jr. 11-Jun-1984 13:58  
Add \$REST\_OF\_LINE macro to HELP verb definition
- V03-004 MHB0148 Mark Bramhall 26-Apr-1984  
Add DISRECONNECT flag.
- V03-003 LY0465 Larry Yetto 22-MAR-1984 13:49  
Add support for rights data base functions.  
This includes the new commands ADD/IDENT, REMOVE/IDENT,  
MODIFY/IDENT, SHOW/IDENT, SHOW/RIGHTS, RENAME/IDENT, LIST/IDENT,  
LIST/RIGHTS
- V03-002 ACG0397 Andrew C. Goldstein, 7-Feb-1984 17:47  
Add DISREPORT flag

V03-001 ACG0388 Andrew C. Goldstein, 12-Jan-1984 18:29  
Add new security features to AUTHORIZE

```
DEFINE TYPE add_generate_type
KEYWORD primary,DEFAULT,NONNEGATABLE
KEYWORD secondary,NONNEGATABLE
KEYWORD both,NONNEGATABLE
KEYWORD all,NONNEGATABLE
KEYWORD current,NONNEGATABLE

DEFINE TYPE generate_type
KEYWORD primary,NONNEGATABLE
KEYWORD secondary,NONNEGATABLE
KEYWORD both,NONNEGATABLE
KEYWORD all,NONNEGATABLE
KEYWORD current,DEFAULT,NONNEGATABLE

DEFINE TYPE ident_attrib_type
KEYWORD resource NEGATABLE

DEFINE TYPE ident_val_type
KEYWORD identifier, NONNEGATABLE, VALUE( REQUIRED, TYPE=$NUMBER )
KEYWORD uic, NONNEGATABLE, VALUE( REQUIRED, TYPE=$INFILE )

DEFINE TYPE primedays_type
KEYWORD monday,NEGATABLE
KEYWORD tuesday,NEGATABLE
KEYWORD wednesday,NEGATABLE
KEYWORD thursday,NEGATABLE
KEYWORD friday,NEGATABLE
KEYWORD saturday,NEGATABLE
KEYWORD sunday,NEGATABLE

DEFINE SYNTAX removeident
ROUTINE uaf$remove_ident
PARAMETER P1, LABEL = token1, PROMPT = "id_name",
VALUE ( REQUIRED )
NOQUALIFIERS

DEFINE SYNTAX removeproxy
ROUTINE remove_proxy
PARAMETER P1,LABEL=token1
PARAMETER P2,LABEL=token2

DEFINE SYNTAX addident
ROUTINE uaf$add_ident
PARAMETER P1, LABEL = token1, PROMPT = "id_name",
VALUE
QUALIFIER attributes, NEGATABLE,
VALUE ( REQUIRED, LIST, TYPE = ident_attrib_type )
QUALIFIER user, NONNEGATABLE, VALUE( REQUIRED, TYPE=$INFILE )
QUALIFIER value, NEGATABLE,
VALUE ( REQUIRED, TYPE = ident_val_type)
```

```

DEFINE SYNTAX addproxy
  ROUTINE add_proxy
  PARAMETER P1,LABEL=token1
  PARAMETER P2,LABEL=token2

DEFINE SYNTAX modifyident
  ROUTINE uaf$modify_ident
  PARAMETER P1, LABEL = token1, PROMPT = "id_name" ,
  VALUE ( REQUIRED )
  QUALIFIER attributes, NEGATABLE,
  VALUE ( REQUIRED, LIST, TYPE = ident_attr_type )
  QUALIFIER holder, NEGATABLE, VALUE ( REQUIRED, TYPE = $INFILE )
  QUALIFIER name, VALUE ( REQUIRED )
  QUALIFIER value, NONNEGATABLE,
  VALUE ( REQUIRED, TYPE = ident_val_type)

DEFINE SYNTAX grantident
  ROUTINE uaf$grant_ident
  PARAMETER P1, LABEL = token1, PROMPT = "id_name" ,
  VALUE ( REQUIRED )
  PARAMETER P2, LABEL = token2, PROMPT = "user id" ,
  VALUE ( REQUIRED, TYPE = $INFILE )
  QUALIFIER attributes, NEGATABLE,
  VALUE ( REQUIRED, LIST, TYPE = ident_attr_type )

DEFINE SYNTAX renameident
  ROUTINE uaf$rename_ident
  PARAMETER P1,LABEL=token1,value(required)
  PARAMETER P2,LABEL=token2,value(required)
  NOQUALIFIERS

DEFINE SYNTAX revokeident
  ROUTINE uaf$revoke_ident
  PARAMETER P1, LABEL = token1, PROMPT = "id_name" ,
  VALUE ( REQUIRED )
  PARAMETER P2, LABEL = token2, PROMPT = "user id" ,
  VALUE ( REQUIRED, TYPE = $INFILE )
  NOQUALIFIERS

DEFINE SYNTAX creatordb
  ROUTINE uaf$create_rdb
  NOPARAMETERS
  QUALIFIER system_id, NEGATABLE, VALUE ( REQUIRED, LIST, TYPE=$NUMBER )

DEFINE SYNTAX createproxy
  ROUTINE create_proxy

DEFINE SYNTAX listident
  ROUTINE uaf$list_ident
  PARAMETER P1, LABEL = token1, PROMPT = "id_name" ,
  QUALIFIER brief, NEGATABLE
  QUALIFIER full, NEGATABLE DEFAULT
  QUALIFIER user, NONNEGATABLE, VALUE( REQUIRED, TYPE=$INFILE )
  QUALIFIER value, NONNEGATABLE,
  VALUE ( REQUIRED, TYPE = ident_val_type)

```

```

DEFINE SYNTAX listproxy
ROUTINE list_proxy
NOPARAMETERS

DEFINE SYNTAX listrights
ROUTINE uaf$list_rights
PARAMETER P1, LABEL = token1, PROMPT = "id_name",
VALUE
QUALIFIER user, NONNEGATABLE, VALUE( REQUIRED, TYPE=$INFILE )

DEFINE SYNTAX showident
ROUTINE uaf$show_ident
PARAMETER P1, LABEL = token1, PROMPT = "id_name",
QUALIFIER brief, DEFAULT
QUALIFIER full, NEGATABLE
QUALIFIER user, NONNEGATABLE, VALUE( REQUIRED, TYPE=$INFILE )
QUALIFIER value, NONNEGATABLE,
VALUE ( REQUIRED, TYPE = ident_val_type)

DEFINE SYNTAX showproxy
ROUTINE show_proxy
PARAMETER P1,LABEL=token1

DEFINE SYNTAX showrights
ROUTINE uaf$show_rights
PARAMETER P1, LABEL = token1, PROMPT = "id_name",
VALUE
QUALIFIER user, NONNEGATABLE, VALUE( REQUIRED, TYPE=$INFILE )

DEFINE SYNTAX mod_sys_pwd,NOPARAMETERS
ROUTINE uaf$mod_sys_pwd

DEFINE TYPE flags_type
KEYWORD audit,NEGATABLE
KEYWORD captive,NEGATABLE
KEYWORD defcli,NEGATABLE
KEYWORD disctly,NEGATABLE
KEYWORD dismail,NEGATABLE
KEYWORD disnewmail,NEGATABLE
KEYWORD disreconnect,NEGATABLE
KEYWORD disreport,NEGATABLE
KEYWORD disuser,NEGATABLE
KEYWORD diswelcome,NEGATABLE
KEYWORD genpwd,NEGATABLE
KEYWORD lockpwd,NEGATABLE
KEYWORD pwd_expired,NEGATABLE
KEYWORD pwd2_expired,NEGATABLE

DEFINE TYPE pflags_type
KEYWORD disnetwork,NEGATABLE
KEYWORD disdialup,NEGATABLE

DEFINE TYPE privileges_type
KEYWORD acct,NEGATABLE
KEYWORD allspool,NEGATABLE
    
```

```

KEYWORD altpri,NEGATABLE
KEYWORD bugchk,NEGATABLE
KEYWORD bypass,NEGATABLE
KEYWORD cmexec,NEGATABLE
KEYWORD cmkrnl,NEGATABLE
KEYWORD detach,NEGATABLE
KEYWORD diagnose,NEGATABLE
KEYWORD downgrade,NEGATABLE
KEYWORD exquota,NEGATABLE
KEYWORD group,NEGATABLE
KEYWORD grpnam,NEGATABLE
KEYWORD grpdrv,NEGATABLE
KEYWORD log_io,NEGATABLE
KEYWORD mount,NEGATABLE
KEYWORD netmbx,NEGATABLE
KEYWORD oper,NEGATABLE
KEYWORD pfnmap,NEGATABLE
KEYWORD phy_io,NEGATABLE
KEYWORD prmceb,NEGATABLE
KEYWORD prmdbl,NEGATABLE
KEYWORD prmjnl,NEGATABLE
KEYWORD prmbx,NEGATABLE
KEYWORD pswapm,NEGATABLE
KEYWORD readal,NEGATABLE
KEYWORD same,NEGATABLE
KEYWORD security,NEGATABLE
KEYWORD setdrv,NEGATABLE
KEYWORD share,NEGATABLE
KEYWORD shmem,NEGATABLE
KEYWORD syslck,NEGATABLE
KEYWORD sysdbl,NEGATABLE
KEYWORD sysnam,NEGATABLE
KEYWORD sysdrv,NEGATABLE
KEYWORD tmpjnl,NEGATABLE
KEYWORD tmpmbx,NEGATABLE
KEYWORD upgrade,NEGATABLE
KEYWORD volpro,NEGATABLE
KEYWORD world,NEGATABLE

```

```

DEFINE VERB add

```

```

ROUTINE add uaf
PARAMETER PT,LABEL=token1,value(required,TYPE=$INFILE)
QUALIFIER access,negatable,value(list)
QUALIFIER account,value(required)
QUALIFIER add identifier, DEFAULT, NEGATABLE
QUALIFIER ast(m,value(required)
QUALIFIER attributes, NEGATABLE,
VALUE ( REQUIRED, LIST, TYPE = ident_attr_type )

QUALIFIER batch,negatable,value(list)
QUALIFIER biolm,value(required)
QUALIFIER bytlm,value(required)
QUALIFIER cli,value(required)
QUALIFIER clitable,value(required)
QUALIFIER cputime,value(required)
QUALIFIER defprivileges,value(required,list,TYPE=privileges_type),
QUALIFIER device,value(required)

```

ONF

ROC

/\*

/\*

/\*

/\*

/\*

agg

enc

enc

```
QUALIFIER dialup,negatable,value(list)
QUALIFIER diolm,value(required)
QUALIFIER directory,value(required,TYPE=$INFILE)
QUALIFIER enqlm,value(required)
QUALIFIER expiration,value(required)
QUALIFIER fillm,value(required)
QUALIFIER flags,value(required,list,TYPE=flags_type)
QUALIFIER generate_password,value(type=add_generate_type)
QUALIFIER identifier, NONNEGATABLE, SYNTAX = addident
QUALIFIER interactive,negatable,value(list)
QUALIFIER jtquota,value(required)
QUALIFIER lgicmd,value(required)
QUALIFIER local,negatable,value(list)
QUALIFIER maxdetach,value(required)
QUALIFIER maxjobs,value(required)
QUALIFIER maxacctjobs,value(required)
QUALIFIER network,negatable,value(list)
QUALIFIER owner,value(required)
QUALIFIER password,value(required,list),NEGATABLE
QUALIFIER pbytlm,value(required)
QUALIFIER pflags,value(required,list,TYPE=pflags_type)
QUALIFIER pgflquota,value(required)
QUALIFIER p_restrict,value(list)
QUALIFIER prclm,value(required)
QUALIFIER primedays,value(required,list,TYPE=primedays_type)
QUALIFIER priority,value(required)
QUALIFIER privileges,value(required,list,TYPE=privileges_type)
QUALIFIER proxy,SYNTAX=addproxy
QUALIFIER pwdexpired,DEFAULT,NEGATABLE
QUALIFIER pdlifetime,value(required)
QUALIFIER pdminimum,value(required)
QUALIFIER quepriority,value(required)
QUALIFIER remote,negatable,value(list)
QUALIFIER sflags,value(required,list,TYPE=pflags_type)
QUALIFIER s_restrict,value(list)
QUALIFIER shrfillm,value(required)
QUALIFIER tgelm,value(required)
QUALIFIER uic,value(required,TYPE=$UIC)
QUALIFIER wsdefault,value(required)
QUALIFIER wsextent,value(required)
QUALIFIER wsquota,value(required)
```

```
DEFINE VERB create
ROUTINE signal_syntax
QUALIFIER proxy, SYNTAX = createproxy
QUALIFIER rightslist, NONNEGATABLE, SYNTAX = createrdb
```

```
DEFINE VERB default
ROUTINE default_uaf
QUALIFIER access,negatable,value(list)
QUALIFIER account,value(required)
QUALIFIER astlm,value(required)
QUALIFIER batch,negatable,value(list)
QUALIFIER biolm,value(required)
```



```

QUALIFIER bytln,value(required)
QUALIFIER cli,value(required)
QUALIFIER clitable,value(required)
QUALIFIER cputime,value(required)
QUALIFIER defprivileges,value(required,list,TYPE=privileges_type),
QUALIFIER device,value(required)
QUALIFIER dialup,negatable,value(list)
QUALIFIER diolm,value(required)
QUALIFIER directory,value(required,TYPE=$INFILE)
QUALIFIER enqlm,value(required)
QUALIFIER expiration,value(required)
QUALIFIER fillm,value(required)
QUALIFIER flags,value(required,list,TYPE=flags_type)
QUALIFIER generate_password,value(type=generate_type)
QUALIFIER interactive,negatable,value(list)
QUALIFIER jtquota,value(required)
QUALIFIER lgicmd,value(required)
QUALIFIER local,negatable,value(list)
QUALIFIER maxdetach,value(required)
QUALIFIER maxjobs,value(required)
QUALIFIER maxacctjobs,value(required)
QUALIFIER network,negatable,value(list)
QUALIFIER owner,value(required)
QUALIFIER password,value(required,list),NEGATABLE
QUALIFIER pbytln,value(required)
QUALIFIER pflags,value(required,list,TYPE=pflags_type)
QUALIFIER pgflquota,value(required)
QUALIFIER p_restrict,value(list)
QUALIFIER p_rclm,value(required)
QUALIFIER primedays,value(required,list,TYPE=primedays_type)
QUALIFIER priority,value(required)
QUALIFIER privileges,value(required,list,TYPE=privileges_type)
QUALIFIER proxy,SYNTAX=addproxy
QUALIFIER pwdexpired,NEGATABLE
QUALIFIER pdlifetime,value(required)
QUALIFIER pdminimum,value(required)
QUALIFIER quepriority,value(required)
QUALIFIER remote,negatable,value(list)
QUALIFIER sflags,value(required,list,TYPE=pflags_type)
QUALIFIER s_restrict,value(list)
QUALIFIER shrfillm,value(required)
QUALIFIER tgelm,value(required)
QUALIFIER uic,value(required,TYPE=$UIC)
QUALIFIER wsdefault,value(required)
QUALIFIER wsextent,value(required)
QUALIFIER wsquota,value(required)

```

```

DEFINE VERB modify
ROUTINE modify uaf
PARAMETER P1,LABEL=token1,value(required,TYPE=$INFILE)
QUALIFIER access,negatable,value(list)
QUALIFIER account,value(required)
QUALIFIER astlm,value(required)
QUALIFIER batch,negatable,value(list)
QUALIFIER biolm,value(required)

```

```

QUALIFIER bytln,value(required)
QUALIFIER cli,value(required)
QUALIFIER clitable,value(required)
QUALIFIER cputime,value(required)
QUALIFIER defprivileges,value(required,list,TYPE=privileges_type),
QUALIFIER device,value(required)
QUALIFIER dialup,negatable,value(list)
QUALIFIER diolm,value(required)
QUALIFIER directory,value(required,TYPE=$INFILE)
QUALIFIER enqlm,value(required)
QUALIFIER expiration,value(required)
QUALIFIER fillm,value(required)
QUALIFIER flags,value(required,list,TYPE=flags_type)
QUALIFIER generate_password,value(type=generate_type)
QUALIFIER identifier, NONNEGATABLE, SYNTAX = modifyident
QUALIFIER interactive,negatable,value(list)
QUALIFIER jtquota,value(required)
QUALIFIER lgicmd,value(required)
QUALIFIER local,negatable,value(list)
QUALIFIER maxdetach,value(required)
QUALIFIER maxjobs,value(required)
QUALIFIER maxacctjobs,value(required)
QUALIFIER modify_identifier, DEFAULT
QUALIFIER network,negatable,value(list)
QUALIFIER owner,value(required)
QUALIFIER password,value(required,list),NEGATABLE
QUALIFIER pbytln,value(required)
QUALIFIER pflags,value(required,list,TYPE=pflags_type)
QUALIFIER pgflquota,value(required)
QUALIFIER p_restrict,value(list)
QUALIFIER prclm,value(required)
QUALIFIER primedays,value(required,list,TYPE=primedays_type)
QUALIFIER priority,value(required)
QUALIFIER privileges,value(required,list,TYPE=privileges_type)
QUALIFIER proxy,SYNTAX=addproxy
QUALIFIER pwdexpired,NEGATABLE
QUALIFIER pdlifetime,value(required)
QUALIFIER pdminimum,value(required)
QUALIFIER quepriority,value(required)
QUALIFIER remote,negatable,value(list)
QUALIFIER sflags,value(required,list,TYPE=pflags_type)
QUALIFIER s_restrict,value(list)
QUALIFIER shrfillm,value(required)
QUALIFIER system_password,value(required),syntax=mod_sys_pwd
QUALIFIER tgelm,value(required)
QUALIFIER uic,value(required,TYPE=$UIC)
QUALIFIER wsdefault,value(required)
QUALIFIER wsextent,value(required)
QUALIFIER wsquota,value(required)

```

```

DEFINE VERB remove
ROUTINE remove_uaf
PARAMETER P1, LABEL = token1, VALUE( REQUIRED, TYPE = $INFILE )
QUALIFIER remove_identifier, DEFAULT, NEGATABLE
QUALIFIER proxy, SYNTAX = removeproxy

```

QUALIFIER identifier, NONNEGATABLE, SYNTAX = removeident

```
DEFINE VERB copy
ROUTINE copy_uaf
PARAMETER P1,LABEL=token1,value(required)
PARAMETER P2,LABEL=token2,value(required)
QUALIFIER access,negatable,value(list)
QUALIFIER account,value(required)
QUALIFIER add_identifier,DEFAULT,NEGATABLE
QUALIFIER astm,value(required)
QUALIFIER attributes,NEGATABLE,
VALUE ( REQUIRED, LIST, TYPE = ident_attr_type )

QUALIFIER batch,negatable,value(list)
QUALIFIER biolm,value(required)
QUALIFIER bytlm,value(required)
QUALIFIER cli,value(required)
QUALIFIER clitable,value(required)
QUALIFIER cputime,value(required)
QUALIFIER defprivileges,value(required,list,TYPE=privileges_type),
QUALIFIER device,value(required)
QUALIFIER dialup,negatable,value(list)
QUALIFIER diolm,value(required)
QUALIFIER directory,value(required,TYPE=$INFILE)
QUALIFIER enqlm,value(required)
QUALIFIER expiration,value(required)
QUALIFIER fillm,value(required)
QUALIFIER flags,value(required,list,TYPE=flags_type)
QUALIFIER generate_password,value(type=generate_type)
QUALIFIER interactive,negatable,value(list)
QUALIFIER jtquota,value(required)
QUALIFIER lgicmd,value(required)
QUALIFIER local,negatable,value(list)
QUALIFIER maxdetach,value(required)
QUALIFIER maxjobs,value(required)
QUALIFIER maxacctjobs,value(required)
QUALIFIER network,negatable,value(list)
QUALIFIER owner,value(required)
QUALIFIER password,value(required,list),NEGATABLE
QUALIFIER pbytlm,value(required)
QUALIFIER pflags,value(required,list,TYPE=pflags_type)
QUALIFIER pgflquota,value(required)
QUALIFIER p_restrict,value(list)
QUALIFIER prclm,value(required)
QUALIFIER primedays,value(required,list,TYPE=primedays_type)
QUALIFIER priority,value(required)
QUALIFIER privileges,value(required,list,TYPE=privileges_type)
QUALIFIER proxy,SYNTAX=addproxy
QUALIFIER pwdexpired,NEGATABLE
QUALIFIER pdlifetime,value(required)
QUALIFIER pdminimum,value(required)
QUALIFIER quepriority,value(required)
QUALIFIER remote,negatable,value(list)
QUALIFIER sflags,value(required,list,TYPE=pflags_type)
QUALIFIER s_restrict,value(list)
QUALIFIER sfrfillm,value(required)
QUALIFIER tqelm,value(required)
```

```
QUALIFIER uic,value(required,TYPE=$UIC)
QUALIFIER wsdefault,value(required)
QUALIFIER wsextent,value(required)
QUALIFIER wsquota,value(required)
```

```
DEFINE VERB rename
ROUTINE rename_uaf
PARAMETER P1,LABEL=token1,value(required)
PARAMETER P2,LABEL=token2,value(required)
QUALIFIER generate_password,value(type=add_generate_type)
QUALIFIER identifier, NONNEGATABLE, SYNTAX = renameident
QUALIFIER password,value(required,list),NEGATABLE
QUALIFIER modify_identifier, DEFAULT
```

```
DEFINE VERB exit
ROUTINE exit_uaf
```

```
DEFINE VERB help
ROUTINE help_uaf
PARAMETER P1, VALUE(TYPE = $REST_OF_LINE)
```

```
DEFINE VERB list
ROUTINE list_uaf
PARAMETER P1,LABEL=token1,VALUE(TYPE=$INFILE)
QUALIFIER brief, DEFAULT
QUALIFIER full
QUALIFIER identifier, NONNEGATABLE, SYNTAX = listident
QUALIFIER proxy, SYNTAX = listproxy
QUALIFIER rights, NONNEGATABLE, SYNTAX = listrights
```

```
DEFINE VERB show
ROUTINE show_user_uaf
PARAMETER P1,LABEL=token1,value(required,TYPE=$INFILE)
QUALIFIER brief
QUALIFIER full,default
QUALIFIER identifier, NONNEGATABLE, SYNTAX = showident
QUALIFIER proxy, SYNTAX = showproxy
QUALIFIER rights, NONNEGATABLE, SYNTAX = showrights
```

```
DEFINE VERB grant
ROUTINE signal_syntax
QUALIFIER identifier, NONNEGATABLE, SYNTAX = grantident
```

```
DEFINE VERB revoke
ROUTINE signal_syntax
QUALIFIER identifier, NONNEGATABLE, SYNTAX = revokeident
```

