English

Fujitsu Software B2000

SECOS V5.6
Security Control System - Audit

User Guide

November 2024

Comments… Suggestions… Corrections…

The User Documentation Department would like to know your opinion on this manual. Your feedback helps us to optimize our documentation to suit your individual needs.

Feel free to send us your comments by e-mail to: bs2000services@fujitsu.com.

Certified documentation according to DIN EN ISO 9001:2008

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2008.

Copyright and Trademarks

Table of Contents

Security Control System - Audit

1 Preface

SECOS (SEcurity COntrol System) comprises a product range of the following individual components: SRPM, GUARDS, GUARDDEF, GUARDCOO, SAT and SECOS-KRB. These components provide administration systems and interfaces with which an individual framework of privileges and responsibilities can be defined for each user. They cover a range of functions extending from setting up, managing and canceling user IDs through working under user IDs to monitoring for any attempts to obtain illegal access to a user ID and its data.

SRPM	(System Resources and Privileges Management). SRPM is used by system administration (and in particular security administrators and user administrators) to define the facilities available to a user ID when this ID is created. The user ID may be linked into a group concept and/or special privileges can be assigned to the user ID. In this manner, system administration sets up a user structure which makes security violations highly improbable and also permits rapid localization of the sources of such violations. The group concept also permits existing project and organization forms to be mapped into the group concept of BS2000.
GUARDS	(Generally Usable Access contRol aDministration System) GUARDS monitors access by the users to files, libraries and other objects belonging to other object administrations. GUARDS protection can be used by object administration for all or each individual user and can be applied to their own objects. GUARDS provides particularly comprehensive and flexible facilities for protecting data against unauthorized access.
GUARDDEF	(Default protection). GUARDDEF is used to allocate default attribute values for files and job variables. Optionally, these values can be prespecified for the creation or modification of these objects. The settings can be made for each pubset by the system administration (TSOS) or by each user for his/her own objects under his/her user ID. GUARDDEF uses GUARDS to store the settings.
GUARDCOO	(Co-owner protection). In the case of files and job variables, a more precise definition of the ownership attribution in the BS2000 (the owner is the ID under which the object is catalogued; TSOS is co-owner of all files and job variables), and which is fixed by default, is possible. It is also possible to withdraw co-ownership for different name ranges associated with the object or for the TSOS user ID or grant it to the TSOS user ID or owners of certain privileges. GUARDCOO uses GUARDS to store the settings.
SAT	(Security Audit Trail). SAT is the logging component of BS2000 for events relevant to security. SAT can be used to identify attempted infiltrations or determine the person at fault in the event of contraventions of the security regulations. For this purpose, SAT logs events in SAT logging files (SATLOG). These files must be evaluated at regular intervals by users who have SAT privileges. This is achieved using the evaluation program SATUT.
	Events which are particularly critical with respect to security can now be monitored without delay with the aid of the new SAT alarm function. The alarm message is displayed on the operator console and the operator can then decide which countermeasures should be implemented.
SECOS-KRB	SECOS-KRB is the interface for handling Kerberos authentication in BS2000.

This manual describes the component SAT (Security Audit Trail).

1.1 Target group

This manual is intended, in particular, for security administration and revisions (evaluation of log files). It descibes the functions of the SECOS component SAT component of the SECOS product. To use this manual, readers will need a good understanding of the security functions present in the BS2000 basic configuration.

1.2 README file

Any additions to the manuals are described in the Readme files for the various product versions. These Readme files are available at http://manuals.ts.fujitsu.com under the various products.

Additional product informations

Current information, version and hardware dependencies and instructions for installing and using a product version are contained in the associated Release Notice. These Release Notices are available at http://manuals.ts.fujitsu.com .

1.3 Changes since the last version of the manual

The changes listed below are only relevant for the SECOS component SAT, which is described in this manual.

Changes with SECOS V5.6

- A new SAT0013 message has been added, which is displayed when the SAT is in HOLD state and slot pool saturation has occurred.
- SAT event UCK for illegal user gives STATION and PROCNAM.
- Default allocation size for SATLOG has been changed to (1002,1002)
- The functionality of the /CHANGE-SAT-FILE command has been extended with the possibility to select any available pubset for storing a SATLOG file.
- The unused option to select a private volume has been removed from the /CHANGE-SAT-FILE command.
- Information of the state of selected SATLOG pubset has been added to /SHOW-SAT-STATUS command output.

1.4 Notational conventions

The following means of representation are used in this manual:

- References to other publications are specified in the form of abbreviated titles in the text. The full title of each publication, to which reference is made by a number enclosed in square brackets, is shown under “Related publications” alongside the relevant number.
- In the examples, user inputs and system outputs ar shown in `fixed-pitch` typeface.
- Special notes on the metalanguage or symbols used only for one SECOS component are provided at the beginning of the related chapter of the manual.
- The metasyntax for SDF commands and statements and the means of representation of command return codes and S variables and macros are explained in the „BS2000OSD/BC - Commands“ manual [4].
- The metasyntax for macros is explained in the „BS2000OSD/BC - Commands“ manual [15].

  This symbol and the word **CAUTION!** precede warning information. In the interests of system and operating security you should always observe this information.

  This symbol denotes important information which you should always observe.

2 SAT - Logging and evaluation of security- relevant data

SAT (Security Audit Trail) supports the logging of security-relevant events in a protected SAT logging file (SATLOG file). The SATLOG file can be analyzed using the SATUT evaluation routine. SATUT edits the SAT logging file and /or generates result lists.

Purposes of the logging of events

- to provide an overview of accesses to objects, to review specific processing steps and actions of particular user IDs and to monitor the use of the security functions
- to detect intrusions into the system by (foreign) users bypassing the security functions
- to detect and prevent any unauthorized use of rights
- to discourage any attempts to bypass the security functions
- to identify the source of a violation of security measures in order to minimize the damage caused
- to initiate an immediate response to unauthorized system intervention (alarm function)

Loggable events

- the use of identification and authentication mechanisms
- the access to objects (e.g. opening of files, program start)
- the creation and deletion of objects
- security-relevant actions of the security administrator, system operation and system administration

Logged data

- date and time of an event
- unequivocal identification of the user; if the chipcard mechanism is used, also identification of the chipcard or the personal user ID
- successful or failed execution of a processing step
- name of the object processed
- description of any modification applied within the framework of user administration or system security measures

The system's CONSLOG files may contain additional events not logged by SAT, e.g. operator replies to questions or actions during BS2000 startup before activation of SAT. Therefore CONSLOG files may be included when evaluating SAT logging.

2.1 Roles and privileges

For security reasons, system administration and system supervision are two areas of activity that should be kept separate. With this in mind, the following roles have been introduced in conjunction with privilege management:

1. The **security administrator**, the user ID with the SECURITY-ADMINISTRATION privilege. The security administrator is responsible for
   - the selection of events (preselection) that are to be stored in the SATLOG files (USER, EVENT, PRESELECTION-RULE, definition of SAT support parameters)
   - the availability of SAT functions (suspending and continuing SAT logging)
   - the definition of events that are to be monitored by the SAT alarm function
   - the definition of filter conditions for refining preselection
   - the assignment of privileges for SAT administration, including the system privileges SAT-FILE-MANAGEMENT and SAT-FILE-EVALUATION

   As delivered, the SECURITY-ADMINISTRATION privilege is permanently assigned to the user ID SYSPRIV. The only possible means of changing this assignment is with the startup parameter service.
2. The **SAT file manager**, the user ID with the SAT-FILE-MANAGEMENT privilege. The file manager is responsible for
   - the management of SAT files, including switching SATLOG files
   - editing events (postselection) that are stored in the SATLOG files with the aid of the SAT evaluation routine SATUT

   As delivered, the SAT-FILE-MANAGEMENT privilege is assigned to the user ID SYSAUDIT. The security administrator can transfer this privilege to any other user ID (except his or her own and TSOS).
3. The **SAT file evaluator**, the user ID with the SAT-FILE-EVALUATION privilege. The file evaluator is allowed to
   - evaluate SATLOG files that have been made available by the SAT file manager.

   As delivered, the SAT-FILE-EVALUATION privilege is assigned to the user ID SYSAUDIT. The security administrator can assign this privilege to a number of different user IDs (apart from his/her own).

   The facility for having reduced SAT logging files evaluated by several user IDs makes it possible to ensure that only specific information about a specific subject (e.g. UTM, file transfer) is evaluated by the administrator of the related product. The security functions of SAT remain the responsibility of the security administrator and the SAT file manager.

2.2 Subject, object and event

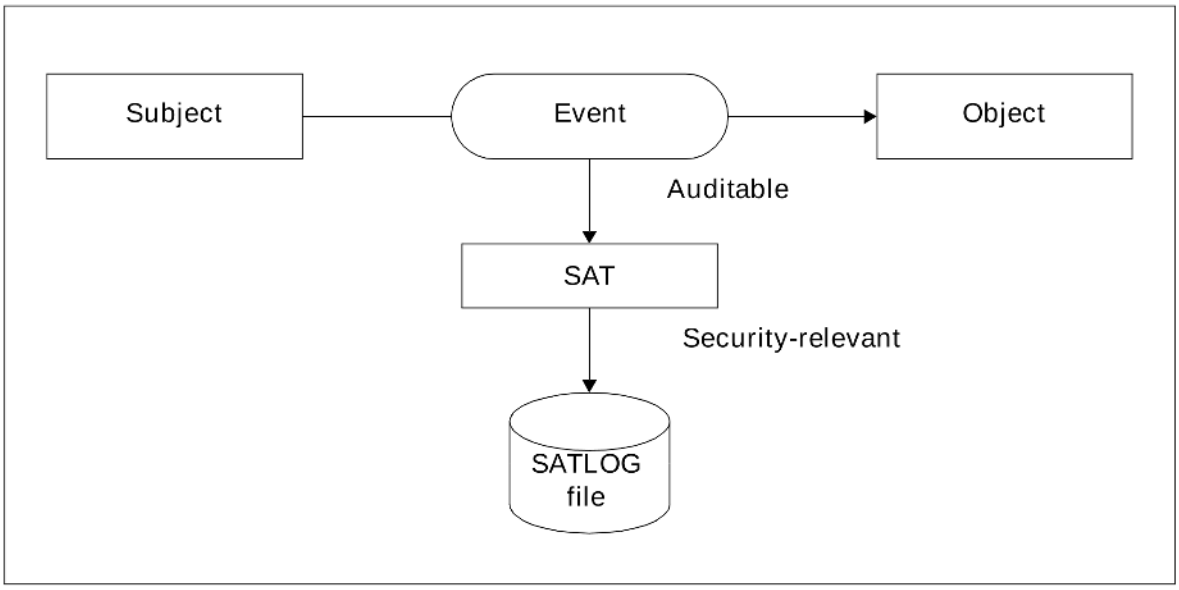

Figure 1: Subject, object and event

A **subject (USER)** is a user of the DP system from which an action such as reading, writing or execution can be initiated. The subject is represented by a user ID.

An **object** is a passive element of a DP system; it contains or receives data and may be subjected to actions such as reading, writing or execution.
In SAT, objects are identified by an object name.
The following are examples of objects:

- files (FILE object name)
- jobs (JOB)
- libraries (PLAM)
- user IDs (USERID)

An **event** is the action of a subject with regard to an object. The result of an event may be “successfully executed (RESULT=SUCCESS)” or “not successfully executed (RESULT=FAILURE)”.
The following are examples of events:

- open a file
- start a job
- activate a subsystem
- export a catalog

An **auditable event (EVENT)** is an event from the list of events that can be logged with SAT. They are identified by a short name, three characters long, such as FMD for “modify file” in the FILE object.
Auditable events are reported to SAT by the system components, with the associated data.

A complete list of objects and auditable events relating to them is given in section “Table of object-related events”. A list of auditable events and the data associated with them is given in section “Tables of auditable information on object-related events (1)”.

A **security-relevant event** is an auditable event to which the selection rules described in section “Selection procedure” apply. Accordingly, an auditable event does not become relevant to security until the links between the audit attributes of the subject, the event and the object indicate the relevance to security.
Security-relevant events are stored by SAT in a SATLOG file, if appropriate after checking by system exit 110, and can be evaluated with SATUT.

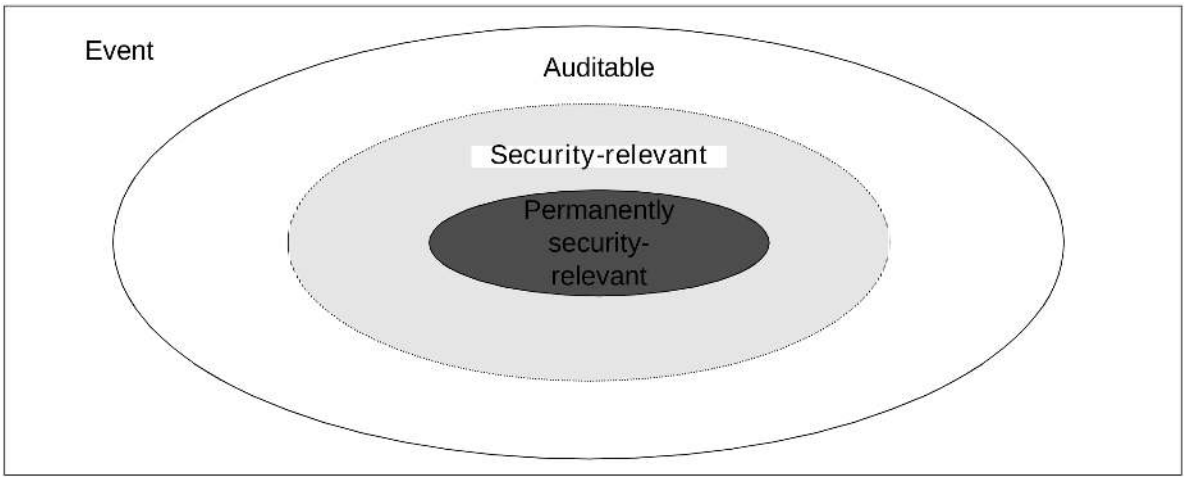

Figure 2: Event types

Permanently security-relevant events are those events which are always of relevance to security when SECOS and SAT are used, with no possibility of change. A default setting is provided for the audit attributes for these events; this setting **cannot** be changed.
The following are permanently security-relevant events:

- actions by the security administrator and the SAT file manager (user ID SYSAUDIT and user IDs with the SECURITY-ADMINISTRATION or SAT-FILE-MANAGEMENT privilege) for the SAT, SAT-ALARM and SAT-FILTER objects
- actions with privileges (granting / withdrawing)

The permanently security-relevant events are identified separately in section “Table of object-related events”.

As regards all other auditable events, the security administrator determines whether they are security-relevant with the aid of the MODIFY-SAT-PRESELECTION command (**preselection**). The security administrator is able to assign the attribute “security-relevant” to an event, and also to withdraw it again.

Some events are considered to be security-relevant when using SECOS and SAT, in addition to the permanently security-relevant events. An audit attribute is defined for these events as a default setting; this can, however, be modified by the security administrator (MODIFY-SAT-PRESELECTION command).
These events and their associated default settings are listed in section “Table of object-related events”.

CONSLOG events

CONSLOG messages are saved in logging files of their own by the operating system. They cannot be evaluated with SAT for logging purposes.
It is possible to include CONSLOG logging files in the evaluation process with the aid of SATUT. To do that, the CONSLOG messages are converted into SATLOG records. The short name for the event type is always CLG for CONSLOG events. However, the contents of the audit record vary depending on which type of CONSLOG message has been converted into a SATLOG record (see "Tables of auditable information on object-related events (1)").

2.3 Controlling logging and evaluation

SAT provides the following optional control functions which enable the volume of data that is dealt with on each specific system to be reduced and which allow appropriate targeting of the execution of evaluations:

1. **SAT support setting**

   This setting, which is made with /MODIFY-SAT-SUPPORT-PARAMETERS, makes it possible to include or exclude events triggered by certain products for logging (and alerting). Currently, this product-specific specification is only available for events triggered by the POSIX product.

   - If SAT support is deactivated for a product then none of the events triggered by this product are logged (and also no SAT alarms are triggered for them). Consequently, steps 2 to 5 below have no effect on the logging of these events.
   - If SAT support is deactivated for a product then steps 2 to 5 below apply without restriction to events triggered by it.
2. **Preselection** – this entails the selection in advance of security-relevant events in SATCP in order to keep the set of events that need to be logged to a minimum.
3. A filter mechanism for refined preselection
4. A system exit, by means of which special cases can be processed selectively.
5. **Postselection** – this entails postprocessing of the saved data with the evaluation routine SATUT for the purpose of selective evaluation and archiving of security-relevant events.
   The results of the evaluation can be output either in replacement files or in analysis files. Replacement files essentially serve the purpose of **archiving** security-relevant information from the input files, and are therefore capable of replacing the input files. In contrast, analysis files are mainly intended for the decentralized **analysis** of security-relevant audit records. Both types of file can be used as input files in a subsequent evaluation run. In addition, edited records can be placed into temporary storage in work files (0 - 9) so that they can be subjected to further processing in the same editing run.

The figure below illustrates the interaction of the control functions 2 to 5 in reducing the possible data volume.
In this case an “event” symbolizes an auditable event that is logged and evaluated by SAT in accordance with the specified selection criteria and rules.

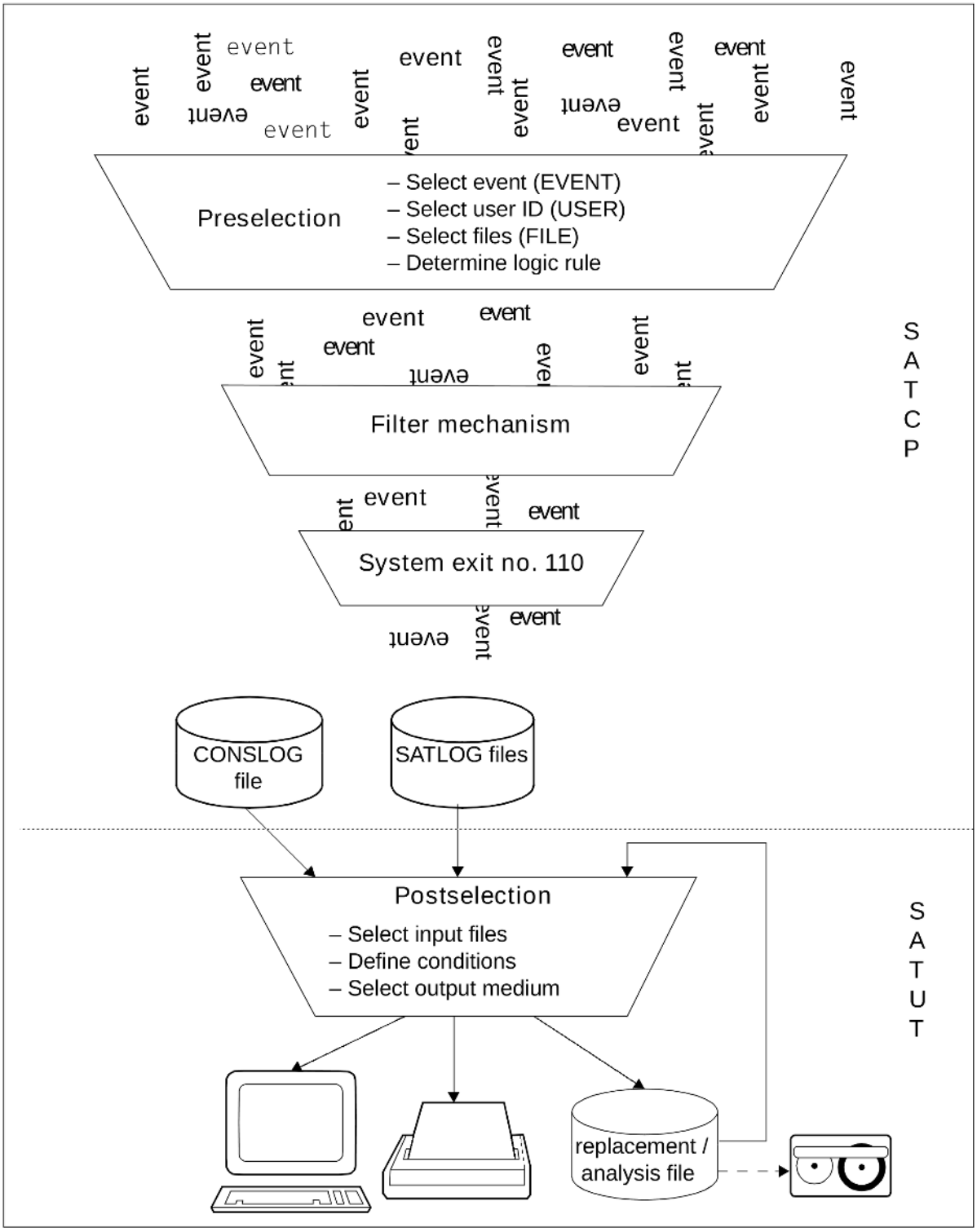

Figure 3: SAT control functions

2.3.1 Selection of security-relevant events (preselection)

The selection of security-relevant events is carried out by the security administrator, in accordance with the following selection procedure.

2.3.1.1 Selection procedure

With the exception of permanently security-relevant events, the security administrator determines which events are security-relevant. If a system is to be operated in accordance with the security standard F2/Q3, the system administrator does not need to make any definitions because the system default setting (see section “Table of object-related events”) conforms to this standard. Should the security administrator require different security criteria, however, he/she can define selection rules for security-relevant events with the /MODIFY-SAT-PRESELECTION command.

The determining elements for selection of a security-relevant event are

- the user ID (USER)
- the auditable event (EVENT) and event result (RESULT)
- the user specifications for the special objects file and library (FILE) and the event result (RESULT)
- the logical operation rule for the above three elements
- the output scope which serves to specify whether *EXTENDED fields are recorded (see section “Tables of auditable information on object-related events (1)”)

The SHOW-SAT-STATUS command can be used by the security administrator and the SAT file manager to display the selection.

User ID (USER)

Security-relevant user IDs are selected by the security administrator by assigning an audit attribute for the user ID with the USER-AUDITING operand of the SAT command /MODIFY-SAT-PRESELECTION.

The following audit attributes can be assigned:

OFF	User ID is not security-relevant
ON	User ID is security-relevant

The audit attribute that is assigned is entered in the user catalog. It takes effect immediately, and remains in effect until it is next changed, even it that is in another session.

Default setting

When SAT is used for the first time, the audit attribute of all user IDs is ON; their auditable events are security-relevant until the security administrator changes the audit attribute. Similarly, the audit attribute for new user IDs that are set up is also ON; their auditable events are security-relevant until the security administrator changes the audit attribute. The command /MODIFY-SAT-PRESELECTION can be used to modify this default setting.
The audit attribute for the SYSAUDIT user ID and user IDs with the SAT-FILE-MANAGEMENT or SECURITY-ADMINISTRATION privilege is also ON, and this cannot be changed; auditable events relating to these user IDs are always security-relevant.

Event (EVENT)

The selection of events is carried out by the security administrator by assigning an audit attribute with the SAT command /MODIFY-SAT-PRESELECTION, EVENT-AUDITING operand.

The following audit attributes can be assigned for the event:

NONE	Event is not security-relevant
SUCCESS	Event is security-relevant if successfully executed (RESULT=SUCCESS)
FAILURE	Event is security-relevant if it is not successfully executed (RESULT=FAILURE)
ALL	Event is security-relevant

Audit attributes for events are recorded in SAT and remain valid in the current session only until they are next changed or until shutdown.
They can also be saved in the SAT parameter file for use in subsequent session (see section “SAT parameter file”).

In the next session the settings in the SAT parameter file apply, whether they are the old settings or modified settings.

Default setting

See section “Table of object-related events”.
The mandatory audit attribute for permanently security-relevant events is ALL; this cannot be changed.

User specifications (FILE)

The audit attribute for the special objects file and library is assigned with the aid of the DMS commands /CREATE-FILE or /MODIFY-FILE-ATTRIBUTES by

- the owner of the file, if authorization to do so has been granted by the group administrator or the global user administrator
  (user catalog entry FILE-AUDIT=*ALLOWED)
- the privileged user TSOS

The following audit attributes can be assigned for file objects:

NONE	Object is not security-relevant
SUCCESS	Object is security-relevant if the event is successfully executed (RESULT=SUCCESS)
FAILURE	Object is security-relevant if the event is not successfully executed (RESULT= FAILURE)
ALL	Object is security-relevant

The audit attribute that is assigned is entered in the file catalog. It takes effect immediately, and remains in effect until it is next changed, even it that is in another session.

Default setting

The file objects are not security-relevant (audit attribute=NONE).

Logic rules

There are two logic rules (logical operation rules) for linking the determining elements for the purpose of selection:

- INDEPENDENT rule
- FILES-BY-EVENTS rule

The logic rule is defined by the security administrator with the PRESELECTION-RULE operand of the /MODIFY-SAT-PRESELECTION command.

The logic rule is recorded in SAT and is initially only valid until the next time that it is changed or until shutdown. It can also be saved in the SAT parameter file for use in subsequent sessions (see section“SAT parameter file”).

In the next session the setting in the SAT parameter file applies, whether the old or a modified setting.

Default setting

INDEPENDENT rule

In the case of the **INDEPENDENT rule** the determining elements are ORed. An event is always logged if at least one of the three determining elements is security-relevant. Accordingly, an auditable event is security-relevant if

- the subject (the user ID) is security-relevant
  i.e. the audit attribute for the user ID is set (ON)

  OR
- the event (EVENT) is security-relevant
  i.e. combination of the audit attributes of EVENT with the event result returns the indicator “security-relevant” (see table).

  OR
- the file object (FILE) is security-relevant
  i.e. combination of the audit attributes of FILE with the event result returns the indicator “security-relevant” (see table).

In the case of objects that are not file objects, FILE is of no relevance and USER OR EVENT applies.

	Audit attribute for EVENT or FILE			
	NONE	**SUCCESS**	**FAILURE**	**ALL**
Event successfully executed **RESULT=SUCCESS**	not security-relevant	security-relevant	not security-relevant	security-relevant
Event not successfully executed **RESULT=FAILURE**	not security-relevant	not security-relevant	security-relevant	security-relevant

Table 1: Combination of the audit attributes of EVENT and FILE with the event result

In the case of the **FILES-BY-EVENTS rule**, EVENT and FILE are ANDed. Accordingly, an auditable event is security-relevant if

- the subject (the user ID) is security-relevant
  i.e. the audit attribute for the user ID is set (ON)

  OR
- the event (EVENT) is security-relevant
  i.e. combination of the audit attribute of EVENT with the event result returns the indicator “security-relevant” (see table above)

  AND
- the file object (FILE) is security-relevant
  i.e. combination of the audit attribute of FILE with the event result returns the indicator “security-relevant” (see table above).

In the case of objects that are not file objects FILE is of no relevance and the condition USER OR EVENT applies, in the same way as for INDEPENDENT logic.

Note

> The make-up of the logic rules shows that even when the set of security-relevant events is reduced to a minimum (see "Individual control of selection") at least **all** auditable events relating to the SYSAUDIT user ID and the user IDs with the SAT-FILE-MANAGEMENT or SECURITY-ADMINISTRATION privileges (see "Selection procedure") are security-relevant and are logged.

Logging quantity

*EXTENDED fields are fields which contain extended information about an event. They are marked in the “Tables of auditable information on object-related events (1)” with an “E”. These fields are only recorded if the security administrator permits recording by specifying LOGGING-QUANTITY=*EXTENDED in the /MODIFY-SAT-PRESELECTION command.

Default setting

*EXTENDED fields are not recorded.

2.3.1.2 Individual control of selection

Default settings

The selection settings for SAT when first used or without individual changes having been made are as follows:

User ID:	for existing user IDs, the selection settings correspond to the entries in the user catalog.
	In the case of newly created user IDs, all the events are logged.
Event:	default setting of security-relevant events (see section “Table ofobject-related events”)
File object:	in accordance with the entries in the file catalog
Logic operation rule:	INDEPENDENT rule
Filter activation:	no filter active
Exit activation:	system exit no. 110 not active
Logging quantity:	*EXTENDED fields are not logged

Selection for the current session

Security-relevant user IDs and events, and the logic rule, and the logging quantity, can be selected by the security administrator with the /MODIFY-SAT-PRESELECTION command, provided that SAT is active.

If SAT is suspended by the security administrator by means of the /HOLD-SAT-LOGGING command and is restarted in the same session with the /RESUME-SAT-LOGGING command the same selection settings apply as before suspension.
If SAT is suspended it is not possible to modify the selection of security-relevant events; the /MODIFY-SAT-PRESELECTION command is not executed.

Selection for subsequent sessions

The security administrator is also able to specify the relevance to security of user IDs and events and the logical operation rule for subsequent sessions.

Settings for user IDs (USER) and file objects (FILE) apply automatically in later sessions because they are stored in the user catalog or file catalog, as appropriate. With regard to events, default settings for new user IDs, the logic rule and the logging quantity the specifications must be stored explicitly in the SAT parameter file with the /SAVE-SAT-PARAMETERS command if they are to take effect the next time the system is started up.

Note on the selection of user IDs

Every **new** user ID is given the audit attribute ON, i.e. all events for these user IDs are automatically logged. If the security administrator does not consider this necessary, he or she can modify this default setting so that all new user IDs receive the audit attribute OFF:

Example

The audit attribute for all new and switchable user IDs is set to OFF (switchable user IDs are all IDs with the exception of SYSAUDIT and IDs with the SECURITY-ADMINISTRATION or SAT-FILE-MANAGEMENT privilege). For the user IDs <user1>, <user2>, <user3>, ... the audit attribute is set to ON. This means that all events that are initiated by the user IDs <user1>, <user2>, <user3>, ... are security-relevant and will be logged.

```
/modify-sat-preselection user-auditing=*default(new-user=*off)
/modify-sat-preselection user-auditing=*all-switchable(audit-switch=*off)
/modify-sat-preselection user-auditing=(<user1>,<user2>,<user3>,...)
```

The user ID TSOS and all user IDs which have been assigned a privilege other than STD-PROCESSING should always be logged. The security administrator’s user IDs, the SYSAUDIT user ID and user IDs with the SAT-FILE-MANAGEMENT privilege are always logged. The logging cannot be switched of for these user IDs. If the privilege SECURITY-ADMINISTRATION (only via startup parameter service) or SAT-FILE-MANAGEMENT (/SET-PRIVILEGE command) is assigned to a user ID whose audit attribute is OFF, then the audit attribute is automatically set to ON.

Note on the selection of events

Specific selection settings that are to take effect on startup can either be stored in the SAT parameter file or they have to be declared again by the security administrator after every system initialization using the /MODIFY-SAT-PRESELECTION command (for example in an automatically executed batch job).

Example

The events “load/execute program” (XLD) and “unload program” (XUL) are to be selected for logging, irrespective of their result. The “add user ID” event (UAD) is to be logged if it is executed successfully, while the “check user ID” event (UCK) is to be logged if it is not executed successfully. Deviating from the default system setting, the security administrator does not consider UTM events (TRM) to be security-relevant and therefore does not want to log them.
These settings are also to apply in subsequent sessions.

The following command is needed to set the selection:

```
/modify-sat-preselection event-auditing=(
                         xld,
                         xul,
                         uad(audit-switch=*on(result=*success)),
                         uck(audit-switch=*on(result=*failure)),
                         trm(audit-switch=*off))
```

To ensure that this setting automatically takes effect on startup, it is possible to execute the command in a batch job that runs with every system startup. Instead of that, however, it is advisable to save the setting in the SAT parameter file after the MODIFY-SAT-PRESELECTION command has been executed once. To save the setting:

```
/save-sat-parameters event-preselection=*current
```

Minimizing the number of logged events

The volume of logged events can be minimized with the aid of the following command:

```
/modify-sat-preselection event-auditing=(                                  -
/ cep(audit-switch=*off),cip(audit-switch=*off),gad(audit-switch=*off),  -
/ gmd(audit-switch=*off),grm(audit-switch=*off),jbe(audit-switch=*off),  -
/ jde(audit-switch=*off),jfk(audit-switch=*off),jin(audit-switch=*off),  -
/ jvg(audit-switch=*off),jvm(audit-switch=*off),jvs(audit-switch=*off),  -
/ kea(audit-switch=*off),ked(audit-switch=*off),kpa(audit-switch=*off),  -
/ kpd(audit-switch=*off),kpm(audit-switch=*off),ktc(audit-switch=*off),  -
/ kxm(audit-switch=*off),mac(audit-switch=*off),psc(audit-switch=*off),  -
/ psd(audit-switch=*off),scr(audit-switch=*off),sct(audit-switch=*off),  -
/ sdl(audit-switch=*off),shd(audit-switch=*off),srm(audit-switch=*off),  -
/ srs(audit-switch=*off),tba(audit-switch=*off),tbd(audit-switch=*off),  -
/ tbe(audit-switch=*off),tbi(audit-switch=*off),tka(audit-switch=*off),  -
/ tkc(audit-switch=*off),tkp(audit-switch=*off),tkr(audit-switch=*off),  -
/ trm(audit-switch=*off),tvm(audit-switch=*off),twk(audit-switch=*off),  -
/ uad(audit-switch=*off),uck(audit-switch=*off),udm(audit-switch=*off),  -
/ uds(audit-switch=*off),uml(audit-switch=*off),ump(audit-switch=*off),  -
/ uop(audit-switch=*off),urm(audit-switch=*off),usl(audit-switch=*off),  -
/ uul(audit-switch=*off),uup(audit-switch=*off),uus(audit-switch=*off),  -
/ vda(audit-switch=*off),vdu(audit-switch=*off),vid(audit-switch=*off),  -
/ vip(audit-switch=*off))
```

This has the effect of disabling logging for all events for which an audit attribute has already been defined as a default setting by BS2000 but which are allowed to be changed (see section “Table of object-related events”).

2.3.2 Refining preselection by the filter mechanism

The filter mechanism allows the security administrator to refine the preselection and thus offers the facility to achieve a targeted reduction in the recording quantity.

> **CAUTION!**
> If recording in accordance with security standard F2/Q3 is required, no filters may be used. The standard preselection must be used.

A maximum of 32 filter conditions can be defined with the following specifications:

- events and result
- subjects (USER-ID)
- information (fields and their contents).

  For field values which can be represented in character form (e.g. <c-string>, <filename>), wildcards may be specified.

These specifications can be made in the form of positive lists (individual listings) or negative lists (*ALL, apart from individual listings). They are logically ANDed to produce a condition. A filter condition thus applies to an audit record whenever all the partial specifications apply to the audit record.

For each filter condition, the TRIGGER-ACTION operand of the /ADD-SAT-FILTER-CONDITIONS or /MODIFY-SAT-FILTER-CONDITIONS command is used to specify an action which is to be performed when the filter condition applies to the audit record.

The following can be specified for TRIGGER-ACTION:

- *LOGGING (RECORDING=*YES)

  The audit record must be recorded when the condition applies.
- *LOGGING (RECORDING=*NO)

  The event is not to be recorded if no other applicable filter condition calls for recording.

An audit record is therefore only recorded whenever all the filter conditions applicable to the record contain the specification TRIGGER-ACTION=*LOGGING(RECORDING=*NO).

If no filter condition applies to an audit record, it will be recorded.

The filter mechanism is controlled by means of the following commands:

ADD-SAT-FILTER-CONDITIONS	Create a filter condition
MODIFY-SAT-FILTER-CONDITIONS	Modify a filter condition
REMOVE-SAT-FILTER-CONDITIONS	Remove a filter condition
SHOW-SAT-FILTER-CONDITIONS	Display a filter condition

The filter definitions can be saved in the SAT parameter file in order that they may be reused during the next session. Definitions which are not explicitly saved lapse on termination of the system session. Saved definitions are automatically activated on commencement of the next system session.

Evaluation of filter conditions

The filter conditions are evaluated following the preselection for the switchable user IDs and the non-permanent security-relevant events which have not already been removed by the preselection. Switchable user IDs are all those user IDs, apart from SYSAUDIT and the user IDs with the privilege SAT-FILE-MANAGEMENT or SECURITY-ADMINISTRATION. Non-permanent security-relevant events are all those events whose audit attribute is changeable (“Y” in the “Audit attribute Chg” column of the “Table of object-related events”ff).

Notes on the performance of the filter mechanism

The filter mechanism offers the facility, through comparison with the information relating to events (fields and their contents), to achieve a targeted reduction in the recording quantity. However, the requisite comparison operations inevitably result in degraded performance in SATCP compared with normal preselection. It is therefore necessary to consider carefully the definition and the utilization of filter conditions.

Activating a filter

A filter is activated immediately after it has been defined (/ADD-SAT-FILTER-CONDITIONS command) and remains active until the end of the system session or until it is deleted by means of the /REMOVE-SAT-FILTER-CONDITIONS command. During this time the definition can be stored, modified or displayed.

2.3.3 Refining selection with system exit no.110

System administration can initiate execution of a SAT exit routing via system exit no. 110. The SAT exit routine makes it possible to suppress the recording of certain auditable events.

A description of the general mode of operation of system exits and a detailed description of system exit 110 is provided in the “System Exits” manual [18].

Execution procedure for system exit 110

Before a SATLOG record is written to the SATLOG file, a copy of the audit record is passed to the system exit together with information about its length. The SATLOG record can be analyzed on the basis of the identifiers for SAT information (see tables starting on "Table of object-related events").

The SAT exit routine can then trigger one of the following, depending on the result of the analysis:

- initiate selective responses (for example blocking a user ID after a certain number of failed LOGON attempts)
- write a separate audit record (ANY event, $SATANY macro)
- on return to SAT via the return code, allow or suppress writing of the analyzed audit record.

Security precautions

The security administrator must explicitly permit the exit routine calls with the `MODIFY-SAT-PRESELECTION ..., EXIT=*YES` command.

Only a copy of the audit record is passed to the exit, thus ensuring that the exit routine is unable to modify the contents of the record.

The exit routine is not invoked for any events for which the logging setting cannot be modified, nor is it invoked for the ANY event.

Exit routines are subsystems with freely selectable names. In a secure system, therefore, system administration should define naming conventions (in particular for system exit no. 110) which unambiguously identify the connection between subsystems and exit routines.

The security administrator has no control over the execution of exit routines. The loading of subsystems should therefore always be monitored in a secure system (in particular for system exit no. 110). To do that, the security administrator must use the /MODIFY-SAT-PRESELECTION command in order to select the events “activate subsystem” (SCR), “hold subsystem” (SHD), “resume subsystem” (SRS) and “deactivate subsystem” (SDL) for logging.

2.3.4 Postprocessing of SATLOG files (postselection)

Editing the SATLOG files is the task of the SAT file manager or the SAT file evaluator. The SATUT utility routine is available for editing purposes, under the SYSAUDIT user ID.
It is executable independently of the SAT subsystem SATCP under any user ID which has the SAT-FILE-MANAGEMENT or SAT-FILE-EVALUATION privilege.
SATUT can also incorporate CONSLOG files in the evaluation, in addition to SATLOG files (see "Tables of auditable information on object-related events (1)").

The SAT evaluation routine SATUT provides the following functions:

- It uses the input files to create edited files (replacement files) containing the securityrelevant data selected by the SAT file manager or SAT file evaluator. In this case the aim is to reduce the volume of data and to store security-relevant audit records, i.e. the input files can be replaced by the edited files.
- It selects specific audit records from the input files on the basis of certain selection conditions. The selected records are output to a printer (SYSLST) or an XML file, or presented in statistical form, or written to a file (analysis file). In this case the aim is to analyze selected event groups, i.e. the input files are not replaced by the edited files.

In order to save storage space for SATLOG files they should be changed regularly (CHANGE-SAT-FILE command), edited as soon as possible, and replaced by replacement files or swapped out to a data medium for long-term archiving. The effect of editing is to reduce the amount of logged data thanks to the more refined selection options provided by SATUT.

The example in the following shows a batch job that was created under the SYSAUDIT user ID and is executed once per day . In this job a job is created that evaluates all SATLOG files and replacement files from the previous day, provided that their names contain a four-digit representation of the year. The result is stored in a single analysis file. Subsequently the SATLOG files used as the input files are automatically deleted, whereas the replacement files are retained.

To prevent unauthorized access to the printed output logs, the output that is made to the system file SYSLST is redirected to a cataloged file. This can be examined on the screen, for example with SHOW-FILE.

```
/SET-LOGON-PARAMETERS
/REMARK *------------------------------------------------------------*
/REMARK * THIS BATCH ANALYZES PRODUCED COLLECTION FILES AND          *
/REMARK * REPLACEMENT FILES BY DAY. ONLY THE OLDEST DAY IS TAKEN     *
/REMARK * INTO ACCOUNT. NO ANALYSIS WILL BE PERFORMED FOR FILES OF   *
/REMARK * THE CURRENT DAY.                                          *
/REMARK *                                                           *
/REMARK * AFTER THE ANALYSIS BY SATUT, ANALYZED COLLECTION FILES WILL*
/REMARK * BE DELETED BUT IN CASE OF SELECTED REPLACEMENT FILES,     *
/REMARK * THE REPLACEMENT FILES ARE NEVER ERASED.                   *
/REMARK *                                                           *
/REMARK * CONDITIONS:  SAT LOGGING HAS TO BE ACTIVE AND ACTIVE      *
/REMARK *              COLLECTION FILE CONTAINS A DATE FORMAT HAVING *
/REMARK *              A DATE CONTAINING A YEAR IN 4 CHARACTERS     *
/REMARK *              (IN THE FILENAME)                            *
/REMARK * INPUT FILES: COLLECTION FILES AND/OR REPLACEMENT FILES    *
/REMARK *              HAVING A DATE CONTAINING A YEAR IN 4         *
/REMARK *              CHARACTERS (IN THE FILE NAME)                *
/REMARK * OUTPUT FILE: ANALYZE.<DATE OF ANALYZED FILES>             *
/REMARK *------------------------------------------------------------*
/ASSIGN-SYSOUT TO-FILE=BATCH.SYSOUT
/ASSIGN-SYSLST TO-FILE=BATCH.SYSLST
```

```
/ASSIGN-SYSDTA TO-FILE=*SYSCMD
/CHANGE-SAT-FILE
/SET-JOB-STEP
/MODIFY-JOB-SWITCHES ON=(4,5)
/START-EDT
@FSTAT 'SYS.SATLOG.////-*' TO 1
@PROC 1
 @@RENUMBER
 @@SET #L2=$
 @@IF #L2 = 1 GOTO 10
 @@NOTE ONLY ONE FILE FOUND -> NO FILES FROM PREVIOUS DAY
 @@SET #L3 = 1
 @@IF #L3:1-21 <> #L2:1-21 GOTO 20
 @@NOTE DATE OF FIRST FOUND FILE EQUAL TO DATE OF LAST FOUND
 @@NOTE FILE -> ALL FILES FROM ACTUAL DAY
@10
 @@SET #S20 = 'NO ANALYZE PROCESSED. SEE REASONS IN THE '
 @@PRINT #S20 NSV
 @@SET #S20 = 'WARNING TEXT.'
 @@PRINT #S20 NSV
 @@RETURN
@20
 @@SET #L1 = 1
 @@ON #L1 FIND 'SYS.SATLOG.'
 @@NOTE EXTRACT DATE
 @@SET #I2 = #I1 + 1

 @@SET #I3 = #I2 + 9
 @@SET #S1 = #L1:#I2-#I3
 @@DELETE
 @@NOTE    CREATE SATUT PROCEDURE
 @@QUOTE !
 @@CREATE  1   : !/SET-LOGON-PARAMETERS!
 @@CREATE  2   : !/ASSIGN-SYSLST TO-FILE=LST.SEL.DAILY.!,#S1
 @@CREATE  2.5 : !/ASSIGN-SYSOUT TO-FILE=OUT.SEL.DAILY.!,#S1
 @@CREATE  3   : !/START-SATUT!
 @@NOTE    SELECT FILES OF PREVIOUS DAY
 @@CREATE  4   : !//SELECT-INPUT-FILES INPUT-FILES=*STD(DATE=!,#S1,!)!
 @@NOTE    SELECT FILES ACCORDING TO CERTAIN CONDITIONS:
 @@CREATE  5   : !//ADD-SELECTION-CONDITIONS NAME=PRIVI, CONDITION= -!
 @@NOTE    SELECT USER IDS
 @@CREATE  6   : !//     OBJ-UID  IN-LIST ('US1','US2','US3') -!
 @@NOTE    DEFINE CONTENTS OF RESULT FIELD
 @@CREATE  7   : !//AND  RES EQUAL F -!
 @@NOTE    DEFINE OBJECT
 @@CREATE  8   : !//AND  EVT IN-LIST ('PST','PRT') !
 @@NOTE    EXECUTE SELECTION
 @@CREATE  8.5 : !//START-SELECTION FROM-FILE=*INPUT-FILES, - !
 @@CREATE  8.7 : !//TO-FILE=*PAR(FILE=0, CONDITION-NAME=PRIVI) !
 @@NOTE    OUTPUT SELECTED RECORDS TO SYSLST
 @@CREATE  9   : !//SHOW-SELECTED-RECORDS SORT-CRITERION=*EVT, -!
 @@CREATE 10   : !//     FROM-FILE=0, OUTPUT=*SYSLST(LINES=114) !
 @@NOTE    SAVE SELECTED RECORDS IN FILE TO BE ARCHIVED
 @@CREATE 11   : !//SAVE-SELECTED-RECORDS TO-REDUCTION-NAME=ANALYZE.!,#S1
 @@CREATE 12   : !//END!
 @@CREATE 12.5 : !/SHOW-FILE-ATTRIBUTES ANALYZE.!,#S1
 @@NOTE    ERASE PROCESSED SATLOG FILES
 @@CREATE 12.7 : !/DELETE-FILE SYS.SATLOG.!,#S1,!.,IGNORE=ACCESS!
 @@CREATE 14   : !/SET-JOB-STEP!
```

```
 @@CREATE 15   : !/ASSIGN-SYSLST TO-FILE=*PRIMARY!
 @@CREATE 15.5 : !/ASSIGN-SYSOUT TO-FILE=*PRIMARY!
 @@CREATE 16   : !/EXIT-JOB SYSTEM-OUTPUT=*NONE!
 @@QUOTE '
 @@WRITE 'E.RUN-DAILY' O
 @@SYSTEM 'ENTER-JOB E.RUN-DAILY'
@END
@DO 1
@HALT
/MODIFY-JOB-SWITCHES OFF=(4,5)
/ASSIGN-SYSDTA TO-FILE=*PRIMARY
/ASSIGN-SYSLST TO-FILE=*PRIMARY
/EXIT-JOB SYSTEM-OUTPUT=*NONE
```

2.3.5 Monitoring special security-relevant activities

Before the security administrator is able to monitor certain security-relevant activities, he or she must first define which events can occur in the course of these activities. This section contains examples of a number of such problem situations. Specifying a preselection reduces the amount of data accruing during the current session.

Examples of the generation of complex condition expressions are given on "ADD-SELECTION-CONDITIONS Define selection conditions".A detailed example of evaluation with SATUT is provided on "Example of evaluation".

Detecting potential intrusion attempts

In order to detect potential intrusion attempts at the time of logon, all failed access attempts are to be evaluated. To achieve this, the security administrator selects the “check user ID” event (UCK) with the result “FAILURE” for logging.

Selection for preselection:

```
/modify-sat-preselection -
/ event-auditing=uck(audit-switch=*on(result=*failure))
```

In order to log failed access attempts, the setting is made in the same way for evaluation (postselection):

```
//add-selection-conditions name=conlog1, -
// condition=evt equal ’uck’ and res equal f
//start-selection from-file=*input-files, -
// to-file=*par(condition-name=conlog1)
```

Detecting file manipulation

File manipulation can be considered to be the successful execution of the following events: “create file” (FCD), “modify file” (FMD), “delete file” (FDD), “rename file” (FRN), “delete protection attributes” (FDS), “convert to decrypted file” (FDC) and “convert to decrypted file” (FEC). They should therefore by selected for logging with the RESULT=SUCCESS.

Selection for preselection:

```
//modify-sat-preselection -
//       event-auditing=(fcd(audit-switch=*on(result=*success)), -
//                       ..., -
//                       fec(audit-switch=*on(result=*success)))
```

Setting for evaluation (postselection):

```
//add-selection-conditions name=confile, -
//    condition=evt in-list (’fcd’,’fmd’,’fdd’,’frn’,’fds’,’fdc’,’fec’) -
//              and -
//              res equal s and filname equal ’<destroyed file name>’
//start-selection from-file=*input-files, -
//                to-file=*par(condition-name=confile)
```

Logging of UTM events

The logging of UTM events (TRM) can be controlled both in SAT and in openUTM.

Selection for preselection:

```
/modify-sat-preselection event-auditing= -
/          trm(audit-switch=<*on/*off>(result=<*all/*success/*failure>), -
/          user-auditing=(<utm-userid1>,<utm-userid2>,...)
```

Setting for evaluation (postselection):

```
//add-selection-conditions name=conutm, -
//                         condition=evt equal 'trm' and <conditions>...
//start-selection from-file=*input-files, -
//                to-file=*par(condition-name=conutm)
```

Control and setting of SAT logging for an UTM application is dealt with by UTM generation and UTM administration. UTM SAT administration is taken care of by UTM users with the appropriate authorization. However, if the SAT logging shall start with the starting of UTM this can only be achieved via the UTM generateion. Generated logging values can be changed with the aid of KDCMSAT.

Detailed information about SAT logging is to be found in the openUTM manual “Generating Applications” [17].

2.3.6 SAT alarm

The SAT alarm function adds an effective checking function to the existing range of SAT functions, allowing immediate detection of violations of security rules or improper behavior during system operation.

Thanks to the SAT alarm function the security administrator is able to detect improper behavior immediately, rather than during subsequent evaluation of the SATLOG files, since a message reporting the violation is output on the system console. This is particularly useful in cases where security violations are committed by users. The classic case of trying out different passwords is an example of such security violations by users.

The alarm function does not replace SAT logging and the evaluation of the SATLOG files, since the violations detected by the alarm function are also entered in the SATLOG file. Furthermore, a large number of alarms resulting from different events will reduce the effectiveness of the alarm. For this reason, the events which are to trigger an alarm should be selected with care.

Whether a SAT alarm is triggered in the form of a message on the console is dependent on

- the event and its result
- the user ID
- information related to the event
- the period within which a certain number of events occurred

The SAT alarm function is controlled with the following commands:

ADD-SAT-ALARM-CONDITIONS	create new alarm definitions
MODIFY-SAT-ALARM-CONDITIONS	modify existing alarm definitions
REMOVE-SAT-ALARM-CONDITIONS	delete existing alarm definitions
SHOW-SAT-ALARM-CONDITIONS	show existing alarm definitions

The alarm definitions can be saved in the SAT parameter file for use in the next session. Definitions which are not explicitly saved are lost when the current session is terminated. Definitions which have been saved are automatically activated again at the beginning of the next session.

Activating an alarm definition

The alarm function is active only when SAT is in recording mode. If SAT is stopped (/HOLD-SAT-LOGGING), no alarm messages will be issued. It is also not possible to enter new alarm definitions or to modify existing definitions while SAT is stopped.

If SAT is in recording mode, an alarm definition becomes active immediately after it has been defined (/ADD-SAT-ALARM-CONDITIONS) and remains active until the end of the session or until it is deleted with /REMOVE-SAT-ALARM-CONDITIONS. In the period between creation and deletion, a definition can be stored, modified or displayed.

If the security administrator has deactivated the connection to SAT logging for a product by means of /MODIFY-SAT-SUPPORT-PARAMETERS then the alarm function for the events relating to this product is inactive (in the current version of SECOS this applies to events relating to the objects “POSIX-FILE-and-Directory”, “POSIX-CHILD-Process”, “POSIX-PROCESS”, “POSIX-SYSTEM-Resources”).

How the alarm function operates

The alarm function is called for every loggable event independently of the preselection. All the defined alarm conditions are then checked to determine whether they apply to the current audit record. An alarm condition is considered to apply to an audit record if all the subconditions it contains are true. A condition which contains a field name is only true if this field is present in the audit record. If a negative list is specified, the condition is true if none of the fields it contains are present in the log record. If all the subconditions in an alarm definition apply to an audit record, a warning is issued at the console.

2.4 Management of SAT

- SAT subsystem SATCP
- SAT parameter file
- SAT logging files (SATLOG)
  - Protection of SATLOG files
  - Changing SATLOG files
  - Storage space requirements
  - Structure of the SATLOG files
- Monitoring by SAT-specific job variable
- Installation and startup

2.4.1 SAT subsystem SATCP

SATCP (SAT Control Program) is that part of SAT that is designed for monitoring events and alarms. The SATCP subsystem is generated and started automatically by DSSM during system startup. This means that SATCP is available before SYSTEM READY. In normal operation, SATCP is active and writes its audit data to the first new SATLOG file of the session. The name of the file is formed by the default name plus the sequence number 1. In addition, the security administrator can interrupt and subsequently resume logging with SAT with the /HOLD-SAT-LOGGING and /RESUME-SAT-LOGGING commands.

If a DMS error prevents the first SATLOG file from being opened, SATCP is still loaded but set to the HOLD state. This is indicated by a warning sent to the console. In that case SAT logging must be restarted with the /RESUME-SAT-LOGGING command.

If a previous session was terminated abnormally, SATCP checks whether the SATLOG files have been closed correctly and verifies their contents if necessary.

The security administrator may suspend SATCP by means of the /HOLD-SAT-LOGGING command; in this case all event logging is stopped and the current SATLOG file is closed. SAT does not then log any events, but when a /RESUME-SAT-LOGGING command is issued it resumes logging with precisely the same parameters as before the /HOLD-SAT-LOGGING command.

The SATCP subsystem is automatically deactivated at system shutdown. It cannot be deactivated by means of the /STOP-SUBSYSTEM command or a similar macro.

2.4.2 SAT parameter file

The SAT parameter file contains all information needed to permit SAT to start with specific settings in the next session. The settings can be made by the user IDs with the privilege SAT-FILE-MANAGEMENT and by the security administrator.

SAT parameters are not automatically stored in the SAT parameter file; this must be done explicitly with the SAVE-SAT-PARAMETERS command (see SAVE-SAT-PARAMETERS command). When the parameters are stored, it is possible to specify which values (*STANDARD or *CURRENT) are to be transferred to the parameter file.

Depending on the privilege possessed by the caller, the following parameters are saved in the SAT parameter file:

- EVENT-PRESELECTION (with the privilege SECURITY-ADMINISTRATION)
- ALARM-CONDITIONS (with the privilege SECURITY-ADMINISTRATION)
- FILTER-CONDITIONS (with the privilege SECURITY-ADMINISTRATION)
- SAT-FILE-ATTRIBUTES (with the privilege SAT-FILE-MANAGEMENT)
- SAT-SUPPORT (with the privilege SECURITY-ADMINISTRATION)

The SAT parameter file $SYSAUDIT.SYSPAR.SAT is created on the HOME pubset as an ISAM file with the attributes:
ACCESS=READ, BLKSIZE=(STD,2), DESTROY=YES and AUDIT=ALL.

The SAT parameter file is opened when the SATCP subsystem is initialized (at STARTUP), and the parameters stored in this file are read. If no SAT parameter file exists at this time, one is created and SAT starts with the default values.

The SAT parameter file remains open as long as the SATCP subsystem is active. All accesses to the SAT parameter file are executed under the control of a system task (SATP task). This is done to prevent unauthorized access to the SAT parameter file.

If an error occurs while opening the SAT parameter file, SAT proceeds as follows. If the error occurred in DMS or in the SATP task environment (SAT subsystem), SAT attempts to recover the SAT parameter file and then tries to open it again. If this cannot be done, the current file is closed (if possible) and the SATP task environment is released. This is indicated by a message on the console. The SAT file manager can then access the SAT parameter file in order to determine the cause of the error and take appropriate steps.

An error in the SAT parameter file does not affect SAT logging or the SAT alarm function. In such an event, however, the default settings are used instead of the values stored in the parameter file.

The error could be eliminated by the following measures:

- The SAT file administrator catalogs the errored SAT parameter file under a new name in order to make it available for diagnostic purposes. Should this not be possible, the administrator deletes the file.
- The SAT file administrator then restarts the system. If no SAT parameter file is found to exist at this time, SATCP creates this using default values.
- Finally, the SAT file administrator sets new current SAT parameter values and saves them by using the /SAVE-SAT-PARAMETERS command and specifying *CURRENT for the desired operands.

Initial installation of SAT

When the system is first started up after initial installation of SAT, there is as yet no SAT parameter file. In this case the SATCP subsystem starts up with the default values and automatically creates a SAT parameter file with those values. In all subsequent sessions SATCP starts up with the values that have been stored in the SAT parameter file.

Changing to a new version

The SAT parameter file contains the SAT parameters for the next session. If a newer version is installed before the next session is started, this newer version recognizes the old format and adapts the old file as follows:

- Old parameters are copied and remain unchanged. The copies are changed to match the new version.
- Newly added parameters are set to their default values.

Note

If the old SAT parameter file contains types (e.g. *STANDARD, *CURRENT), rather than concrete values, as operand values, then these are transferred without change to the new version. SAT does not check whether these types have different meanings in the new version.

Examples of SAT parameter files

Changing the SAT parameter file

In the current session values for event selection have been changed with the /MODIFY-SAT-PRESELECTION command, and the file attributes of the SATLOG file have been changed by means of the /CHANGE-SAT-FILE command. The alarm and filter conditions as well as the SAT support parameters have not been changed in this session.

The **security administrator** stores the currently valid values for event selection and the alarm function in the SAT parameter file with the following command:

```
/save-sat-parameters event-preselection=*current, -
/                    alarm-conditions=*current, -
/                    filter-conditions=*current, -
/                    sat-support=*current
```

The next session would begin with the following settings:

EVENT-PRESELECTION	Values valid when the SAVE-SAT-PARAMETERS command was input
ALARM-CONDITIONS	Values as in the last session, because they were not changed during execution of the SAVE-SAT-PARAMETERS command
FILTER-CONDITIONS	Values as in the last session, because they were not changed during execution of the SAVE-SAT-PARAMETERS command
SAT-FILE-ATTRIBUTES	Values as in the last session, because the values set with /CHANGE-SAT-FILE were not stored
SAT-SUPPORT	Values as in the last session, because they were not changed during execution of the SAVE-SAT-PARAMETERS command

The **SAT file manager** would also like to transfer the changes made to the attributes of the logging file to the SAT parameter file. This is done with the following command:

```
/save-sat-parameters sat-file-attributes=*current
```

The next session therefore begins with the following settings:

EVENT-PRESELECTION	as above
ALARM-CONDITIONS	as above
FILTER-CONDITIONS	as above
SAT-SUPPORT	as above
SAT-FILE-ATTRIBUTES	Values valid when the SAVE-SAT-PARAMETERS command was input

2.4.3 SAT logging files (SATLOG)

Each SATLOG file consists of audit records which describe security-relevant events.

The SATLOG files are created on the user-specified pubset under the SYSAUDIT user ID, with a name structured as follows:

$SYSAUDIT.SYS.SATLOG.yyyy-mm-dd.sss.nn where:

yyyy-mm-dd	creation date
sss	session number
nn	number of the file in this session (01 through 99)

The FMTYFNLG system parameter does not affect SATLOG files. The SATLOG files are created as SAM files with the attribute DESTROY=YES in EXTEND mode, with a block size of (STD,2), space allocation (1002,1002) and audit attribute ALL.

These records are written by a separate task for performance reasons. SAT makes use of the CLTF (Common Log Task Facility) interface for this purpose.

2.4.3.1 Protection of SATLOG files

Any user who has the privilege SAT-FILE-EVALUATION or SAT-FILE-MANAGEMENT may evaluate SATLOG files. These files must be protected in such a way that there are accessible only from the user IDs which possess the privilege SAT-FILE-EVALUATION or SAT-FILE-MANAGEMENT. Optimum protection can be achieved by linking the SATLOG files and SAT reduction files to a guard. This guard can then contain conditions which permit access to the SAT files only with a specific privilege and only with a specific program.

In addition, the audit attribute is set, i.e. any access to the SAT files is logged automatically (with the logic rule INDEPENDENT). This covers the opening, closing and replacing of SAT files.

The following rules should be observed, in particular when the SAT-FILE-MANAGEMENT or SAT-FILE-EVALUATION privilege is assigned to user IDs other than SYSAUDIT:

- The SATLOG files cataloged under SYSAUDIT cannot be evaluated unless they are shareable. SRPM group administration should therefore create an AUDITOR group of which the user ID SYSAUDIT is a member. The files should be made accessible to this group by means of a basic access control list (BACL) or a guard.
- The deletion of SATLOG files and the creation of replacement files are possible only under the user ID SYSAUDIT. Analysis files or lists can be created under other user IDs. (see section “Input files for SATUT”).

2.4.3.2 Changing SATLOG files

The security administrator and the SAT file manager are able to change SATLOG files with the CHANGE-SAT-FILE command. When doing so, the SAT file manager can also move SATLOG files to any available pubset and change the file attributes. When SATLOG files are closed, the catalog entry can be optimized with ACCESS=READ and SPACE=RELEASE(-9999).

Explicit changing

The current SATLOG file can be changed manually with a command if it becomes too large or if a change is needed for any other reason. SAT file management can use the CHANGE-SAT-FILE command to close the current SATLOG file and open a new one without any loss of information.

Implicit changing

The current SATLOG file is replaced implicitly - i.e. without involving SAT file management - whenever it is affected by a DMS error (including storage saturation). With certain DMS errors (e.g. ’no disk space available’, i.e. when the primary storage allocation has been used up and the secondary allocation is zero), it may not be possible to write the trailer record.

Periodic changing

The current SATLOG file can be changed automatically at fixed intervals with the aid of the REPEAT operand of the /CHANGE-SAT-FILE command. No data is lost in the process.Each time period begins with entry of the /CHANGE-SAT-FILE command, or by the /RESUME-SAT-LOGGING command if logging with SAT had been suspended.

2.4.3.3 Storage space requirements

Storage space requirements for SAT logging increase with the number of events to be audited. The number of events to be audited can be reduced if preselection of the events for auditing is carried out (see section “Selection procedure”). The definition of selection criteria thus enables the security administrator to exercise a considerable influence on the storage space requirements for SAT.

The volume of data actually logged depends on the size and workload of the installation and the range of applications involved. It is sound practice to determine the appropriate space allocation by means of test runs or with the aid of the SATUT statement //SHOW-STATISTICS (see SHOW-STATISTICS command).

The following example should provide a rough idea of how this works.

In the case of the default logging setting (i.e. no /MODIFY-SAT-PRESELECTION command was issued) and the audit attribute NONE for all FILE objects, the following average values were determined:

Length of an audit record: 75-80 bytes
Number of events logged: 700-800 events/MIP/hr
Required storage space: 30-35 PAM pages/MIP/hr

Above all the audit attributes of the FILE objects (see section “Selection procedure”) can influence the SAT storage space requirements, as events relating to FILE objects account for approximately 50% of all possible events. These events are logged in accordance with their results and audit attributes.

Storage saturation

In the event of storage space problems (no more space on public volumes), a corresponding DMS error prevents continuation of SAT logging. In order to prevent audit data being lost, SAT suspends the jobs that intended to write an audit record (macro VPASS see the “Executives Macros” manual [15]). Logging continues only for the users with the privileges SECURITY-ADMINISTRATION and SAT-FILE-MANAGEMENT, since their audit records are stored in class 5 memory until such time as the situation has returned to normal. If a /EXIT-JOB (resp. /LOGOFF) command is issued, the audit records of these user IDs are not lost and the command is executed as follows:

- the connection to the terminal is closed
- the task is not terminated until SATCP returns to logging mode
- /SET-LOGON-PARAMETERS (resp. /LOGON) is rejected for nonprivileged users, that is to say all users except for those who have the SECURITY-ADMINISTRATION, SAT-FILE-MANAGEMENT or OPERATING privilege.

If the SAT status switches to NO-RESOURCE, a message indicating the nature of the problem involved is displayed at the console. The security administrator and SAT file management can take appropriate measures and logging is continued automatically.

Example

For “disk space saturation”:

```
/change-sat-file ... ,support=*public(...)
```

Jobs whose SATLOG records have been stored in class 5 memory while SATCP was in NO-RESOURCE status will under certain circumstances remain suspended even after the /HOLD-SAT-LOGGING command has been entered (VPASS macro, see the “Executives Macros” manual [15]). In this case, SATCP outputs a message at regular intervals which indicates this status. In order to resume processing the suspended jobs, it is additionally necessary to enter the /RESUME-SAT-LOGGING command.

2.4.3.4 Structure of the SATLOG files

All audit records have the same structure. They consist of a list of fields, each field containing an item of auditable information.

If nothing other than SAT commands and SATUT statements is used, no knowledge of the structure of the SATLOG files is required.
The information given in the following is only important if system exit No. 110 is used or when explicit file analysis is performed in the event of an error.

Note

The structure of the records is dependent on the version, and is described by the macro EXIT110 (see the “System Exits” manual [18]).

Header records / trailer records

Audit records are prefixed / suffixed by SAT-specific records containing information relating to the special events “start / end of SATLOG file”. These specific records are created directly by the file handling and are not forwarded to CLIP.

The header record (ZBG) contains the following:

- system version
- system name
- reason for creating the file (startup, resume logging...)
- name of the preceding SATLOG file (if any)
- CPU identification
- system identification
- name of the configuration

The trailer record (ZND) contains the following:

- name of the next SATLOG file of this session
- reason for closing the file (shutdown, change file...)

Records

The fields of each audit record are arranged as follows:

- The first part of the record is invariable and contains the fields/items of information that are always logged for any record.

Field name	Length	Meaning	
user-id	8	User ID of the subject	Invariable part Length 28 characters
tsn	4	TSN of the subject	
evt	3	Abbreviated event name	
res	1	Result of the event (S/F)	
	4	Date of creation Format: X‘yyyymmdd‘	
	4	Time of creation Format: X‘hhmmss00‘	
	4	Reserved area	

- The second part of the record is variable. It includes the fields/items of information which may but need not be logged for any record (e.g. auditid, groupid), as well as the fields/items of information logged for specific objects only (see section “Tables of auditable information on object-related events (1)”). These are variable-length fields. Each field in the variable part contains the actual length of the information, the exit identifier for the SAT information and the information itself

ln1	id1	<info1>		variable part variable length
ln2	id2	<info2>		
ln3	id3	<info3>		
...	...	...		

ln n:

Length of the logged information <info n> of field n (1 byte)

In the case of *LNG fields, this field contains the value 255.

id n:

Exit identifier for the SAT information contained in field n (2 bytes)

In the case of *LNG fields, this field contains the negative value of the exit identifier.

info n:

Logged information of field n (field value, in n bytes); keywords are binary-coded.

**LNG fields*

*LNG fields are fields whose length exceeds 255 characters. If necessary they are split over several audit records and are structured as follows:

255	- id	ln	0	<info1>		First or only audit record for the *LNG field
255	- id	ln	displ	<info n>		Continuation record for a *LNG field if this is split over several audit records

ln:

Overall length of logged information for the *LNG field (2 bytes).

- id:

Exit identifier for the SAT information which is contained in a *LNG field (2 bytes, value is negative).

displ:

Displacement of first byte of subsection “info n” from the start of the total information (2 bytes)

info n:

n-th section of the logged information of the *LNG field

The maximum length of a SATLOG record in the SATLOG file is 1000 bytes. The unsplit SATLOG record is made available to the EXIT routine. The maximum length is 32752 bytes, *displ* is always 0.

2.4.4 Monitoring by SAT-specific job variable

The status of the SAT logging and the name of the current SATLOG file can be monitored using a job variable. This job variable has the fixed name `:<home-catid>:$SYSAUDIT.SYS.SAT.SATLOG-FILENAME`. It is assigned values by SAT whenever the SATLOG file is changed and whenever the SAT status changes. If it does not exist it is automatically created under the SYSAUDIT ID on the home pubset.

From the start of a system run up to the first change of the SATLOG file the content of the job variable is undefined.

The structure of the job variable corresponds to that of job-monitoring job variables (see the “Job Variables” manual [31]). The following entries have SAT-specific meanings:

Byte	Meaning/Possible values
1-3	SATstatus: $R: RECORD (recording mode) $H: HOLD (i.e. the /HOLD-SAT-LOGGING command was executed) $N: NO RESOURCE $S: SHUTDOWN
17	Type of MONJV: “A”
71-128	Name of the SATLOG file In recording mode (SAT status= “$R”) the current SATLOG file is concerned, in all other cases the SATLOG file that has just been closed.

2.4.5 Installation and startup

The installation procedure for SAT comprises installing the software for the SATCP subsystem and for the SATUT utility routine. Before SAT is put into operation you should draw up a plan for logging security-relevant data on the basis of the SAT control options. For further details refer to section “Selection of security-relevant events (preselection)”, section “Postprocessing of SATLOG files (postselection)” and „section“Monitoring special security-relevant activities”.

Installing SATCP

The following files must be cataloged under TSOS in order to install SAT:

File	File name
Subsystem catalog	$TSOS.SYSSSC.SATCP.nnn
Subsystem library	
... for SU_/390 and S servers	$TSOS.SYSLNK.SATCP.nnn
... for SU x86 and SQ servers	$TSOS.SKMLNK.SATCP.nnn
Syntax file	$TSOS.SYSSDF.SATCP.nnn
Message file	$TSOS.SYSMES.SATCP.nnn
Rep file	$TSOS.SYSRMS.SATCP.nnn
IMON file	$TSOS.SYSSII.SATCP.nnn
ENTER job for the transfer of the preselection when a new AT version is used	$TSOS.SYSSSC.SATCP.nnn

Table 2: Installation files for SATCP (nnn = version of subsystem)

The SATCP subsystem is activated and started automatically by DSSM during system startup. This means that SAT is available prior to “SYSTEM READY”. In normal operation, SAT is active and writes its audit data to the first new SATLOG file of the session. The SATCP subsystem is implicitly deactivated during system shutdown; it cannot be deactivated explicitly.

In a BS2000 system in which SECOS is being used for the first time, all the user IDs have the log setting AUDIT-SWITCH=*ON after initial installation.

In a BS2000 system which has been upgraded from a lower to a higher SECOS version, all the user IDs retain their previous log setting after installation.

An ENTER procedure is available to back up the current preselection settings for the user IDs prior to any change. This generates an ENTER job with the corresponding /MODIFY-SAT-PRESELECTION commands.

Installing SATUT

The following files are supplied with SATUT :

File	File name

Module library	$**SYSAUDIT**.SYSLNK.SATUT.nnn
System syntax file	$TSOS.SYSSDF.SATUT.nnn
Message file	$TSOS.SYSMES.SATUT.nnn
Rep file	$TSOS.SYSRMS.SATUT.nnn
IMON file	$TSOS.SYSSII.SATUT.nnn

Table 3: Installation files for SATUT (nnn = version of subsystem)

SATUT runs independently of SAT under any user ID with the privilege SAT-FILE-MANAGEMENT or SAT-FILE-EVALUATION.

2.5 SAT commands

The following sections first provide a functional overview of all SAT commands and then go on to describe the individual commands in alphabetical order. Each command description starts with a general explanation of the function of the command, followed by the command format and a description of the various operands and their values. The operand description is followed by the command return code and, where appropriate, an example of application.

The command metasyntax is explained in the appendix.

2.5.1 Functional overview

Product-specific activation and deactivation of logging and alarms

MODIFY-SAT-SUPPORT-PARAMETERS	Activate or deactivate SAT logging and SAT alarms for a given product
SHOW-SAT-SUPPORT-PARAMETERS	Display the product for which SAT logging and the triggering of SAT alarms is activated

SAT preselection

MODIFY-SAT-PRESELECTION	Change the selection of events (EVENT-AUDITING) and user IDs (USER-AUDITING), change the logical operation rule (PRESELECTION-RULE), change the authorization to invoke system exit 110 (EXIT) and define the logging quantity
SAVE-SAT-PARAMETERS	Save the specified SAT parameters for the next system start in a SAT parameter file
SHOW-SAT-STATUS	Display the selected user IDs, events and the set logical operation rules (... INFORMATION=*USER-AUDITING / *EVENT-AUDITING / *PRESELECTION-RULE)

SAT filter mechanism

ADD-SAT-FILTER-CONDITIONS	Definition of conditions for preselection for SAT logging
MODIFY-SAT-FILTER-CONDITIONS	Modify filter conditions
REMOVE-SAT-FILTER-CONDITIONS	Delete filter conditions
SAVE-SAT-PARAMETERS	Save the specified SAT parameters for the next system start
SHOW-SAT-FILTER-CONDITIONS	Display the filter conditions.

Controlling SAT logging

HOLD-SAT-LOGGING	Suspend auditing and close the SATLOG file
RESUME-SAT-LOGGING	Resume auditing in a new SATLOG file
SHOW-SAT-STATUS	Display the status of SAT logging (... INFORMATION=*LOGGING-STATUS)

SATLOG file

CHANGE-SAT-FILE	Close the current SATLOG file and open a new SATLOG file. No data is lost during the changeover.
SAVE-SAT-PARAMETERS	Save the specified SAT parameters for the next system start in a SAT parameter file

SHOW-SAT-STATUS	Display the name of the SATLOG file, the attributes of the file and the current logging status (... INFORMATION=*COLLECTION-FILE)

SAT alarm function

ADD-SAT-ALARM-CONDITIONS	Definition of conditions which trigger a console message to inform the security administrator immediately of any security violations
MODIFY-SAT-ALARM-CONDITIONS	Modify alarm conditions
REMOVE-SAT-ALARM-CONDITIONS	Delete alarm conditions
SAVE-SAT-PARAMETERS	Save the specified SAT parameters for the next system start
SHOW-SAT-ALARM-CONDITIONS	Display the alarm conditions.

Notes

The above commands are available in SATCP with various functional scopes, depending on the logging status:

Command	**SATCP status (as shown by SHOW-SAT-STATUS)**		
	HOLD	**RECORD**	**NO-RESOURCE**
HOLD-SAT-LOGGING	-	X	X
RESUME-SAT-LOGGING	X	-	-
MODIFY-SAT-PRESELECTION	-	X	-
SHOW-SAT-STATUS	X	X	X
MODIFY-SAT-SUPPORT-PARAMETERS	-	X	-
SHOW-SAT-SUPPORT-PARAMETERS	X	X	X
CHANGE-SAT-FILE	-	X	-
SAVE-SAT-PARAMETERS	-	X	-
ADD-SAT-ALARM-CONDITIONS	-	X	-
MODIFY-SAT-ALARM-CONDITIONS	-	X	-
REMOVE-SAT-ALARM-CONDITIONS	-	X	-
SHOW-SAT-ALARM-CONDITIONS	X	X	X
ADD-SAT-FILTER-CONDITIONS	-	X	-

MODIFY-SAT-FILTER-CONDITIONS	-	X	-
REMOVE-SAT-FILTER-CONDITIONS	-	X	-
SHOW-SAT-FILTER-CONDITIONS	X	X	X

All SAT commands are serialized, i.e. a SAT command will be rejected if any other command is currently executing (exception: the commands /SHOW-SAT-STATUS, /SHOW-SAT-ALARM-CONDITIONS, /SHOW-SAT-FILTER-CONDITIONS and /SHOW-SAT-SUPPORT-PARAMETERS, in all cases with the operand VALUE=*STD/*CURRENT).

If an error occurs during execution of a SAT command, the spin-off mechanism is triggered.

2.5.2 ADD-SAT-ALARM-CONDITIONS Define alarm conditions

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The security administrator uses this command to define conditions for the occurrence of an alarm situation. The alarm definition can be displayed by the /SHOW-SAT-ALARM-CONDITIONS command. It can be removed again by means of /REMOVE-SAT-ALARM-CONDITIONS.

The events which are to trigger alarms are specified as follows:

- by the event name and the result on occurrence of the event
- by the user ID of the recorded event
- by the information relating to the event

If a certain number of such events occur within a specified period of time, an alarm is triggered in the form of a message on the operator console.

ADD-SAT-ALARM-CONDITIONS

```
NAME = <name 1..8>
, SELECT = *PARAMETERS(...)
    *PARAMETERS(...)
        |  EVENT-NAME = *ALL / list-poss(50): <name 3..3>(...)
        |      <name 3..3>(...)
        |          |  RESULT = *ALL / *SUCCESS / *FAILURE
        |  , USER-IDENTIFICATION = *ALL / list-poss(50): <name 1..8>
        |  , FIELD-NAME = *ALL / list-poss(50): <name 3..7>(...)
        |      <name 3..7>(...)
        |          |  VALUE = *ALL / *MATCH(...) / *NOT-MATCH(...) / list-poss(10): <text> /
        |          |                    list-poss(10): <integer 0..2147483647>(...)
        |          |      *MATCH(PATTERN=<text>)
        |          |      *NOT-MATCH(PATTERN=<text>)
        |          |      <integer 0..2147483647>(...)
        |          |          |  UNIT = *BYTES / *KB / *MB / *GB
, TIME-LIMIT = *UNDEF INED / *WITHIN(...)
    *WITHIN(...)
        |  DAYS = <integer 0..365>
        |  , HOURS = <integer 0..23>
        |  , MINUTES = <integer 0..59>
, REPEAT = 3 / <integer 1..255>
, TRIGGER-ACTION = *OPERATOR-MESSAGE(...)
    *OPERATOR -MESS AGE(...)
        |  WAIT-RESPONSE = *YES / *NO
```

NAME = <name 1..8>
Name of the alarm.

SELECT = *PARAMETERS(...)
This defines which conditions must be fulfilled in order to trigger the action specified for the TRIGGER-ACTION operand of this command.

EVENT-NAME =
Type and result of the event(s) to be monitored.

EVENT-NAME = *ALL

All events which can be recorded by SAT are to be monitored for the alarm function.

EVENT-NAME = list-poss(50): <name 3..3>(...)
The explicit name of an event. This name must be taken from "Table of object-related events". If you specify POSIX events, please pay special attention to Note 4.

RESULT = *ALL / *SUCCESS / *FAILURE
Specifies the result the event is to have.

USER-IDENTIFICATION =
The user IDs which are to be monitored.

USER-IDENTIFICATION = *ALL
All user IDs are to be monitored.

USER-IDENTIFICATION = list-poss(50): <name 1..8>
The specified user IDs are to be monitored. The user ID does not need to exist at the time when the alarm condition is defined.

FIELD-NAME =
This specifies which field of an event is to be monitored.

FIELD-NAME = *ALL
All fields of an event are to be monitored.

FIELD-NAME = list-poss(50): <name 3..7>(...)
Only a field specified here are to be monitored. A list of the possible field names can be found in "Tables of auditable information on object-related events (1)".

VALUE = *ALL / *MATCH(...) / *NOT-MATCH(...) / list-poss(10): <text> / list-poss(10): <integer 0..2147483647>(...)

A list of the field names and the information output in these fields can be found in "Tables of auditable information on object-related events (1)".
<text> depends on the field being logged.

VALUE = *MATCH(...)
Specifies a pattern for the field name. The condition is valid if the value for comparison matches this pattern. The pattern specification is only permitted for field names whose values represent a string (<c-string>, <filename>, <name>).

PATTERN = <text>
Pattern specification in the format <c-string 1..255> in which, similarly to the SDF data type <c-string with-wild (n)>, parts of the string can be replaced by wildcards.

The available wildcard characters are as follows:

*	Stands for any desired character string, including a blank string
/	Stands for precisely one character
\	Nullifies the effect of “wildcards” (* / < > : ,) actually forming part of the character string (e.g. ab*c denotes the actual character string “ab*c”)
<s_x:s_y>	Replaces a character string where the following applies: • it is at least as long as the shortest character string (s_x or s_y) • it is at most as long as the longest character string (s_x or s_y) • it falls between s_x and s_y in the alphabetical sort sequence; numbers are sorted after letters (A...Z 0...9)
	• s_x may also be the blank character string which appears at the beginning of the alphabetical sort sequence • s_y may also be the blank character string which stands at this position for the character string with the highest possible coding (contains only the characters X‘FF‘) • s_x must precede s_y in the alphabetical sort sequence. If s_x is shorter than s_y, s_x will be padded with X‘00‘ • if s_y is shorter than s_x, s_y will be padded with X‘FF‘ • no wildcards may occur either in s_x or in s_y
<s1,...>	Replaces all character strings to which one of the character combinations specified by s applies. s may also be a blank character string. Any character string s may also be a range specification <s_x:s_y>

VALUE = *NOT-MATCH(...)
Specifies a pattern for the field name. The condition is valid if the value for comparison does **not** match this pattern. The pattern specification is only permitted for field names whose values represent a string (<c-string>, <filename>, <name>).

PATTERN = <text>
Pattern specification as in VALUE=*MATCH.

VALUE = <integer 0..2147483647>(...)
Specifies a numerical value for the field name. This value is only allowed for fields whose value is of type <integer>.

UNIT = *BYTES / *KB / *MB / *GB
Specifies the units to be used in interpreting the value specified with the VALUE operand. This entry is only allowed for field names filpos, curlim2 and maxlim2.

The following thereby applies:

- If UNIT=*BYTES is implicitly or explicitly defined, the value must be a multiple of 512.
- The maximum value of 2^{40}-512 (=1 099 511 627 264) bytes may also not be exceeded if UNIT=*KB / *MB / *GB is specified. This results in the following maximum values, depending on the UNIT entry:

UNIT=	Maximum value for VALUE	Corresponds in bytes to
***BYTES**	2^{31}-1 = 2 147 483 647	2^{31}-1 = 2 147 483 647
***KB**	2^{30}-1 = 1 073 741 823	2^{40}-2^{10} = 1 099 511 626 752
***MB**	2^{20}-1 = 1 048 575	2^{40}-2^{20} = 1 099 510 579 200
***GB**	2^{10}-1 = 1 023	2^{40}-2^{30} = 1 098 437 885 952

TIME-LIMIT =
The period within which x (defined with REPEAT) occurrences of an event are to trigger an alarm.

TIME-LIMIT = *UNDEFINED
The entire period of SAT logging is to be evaluated. This means that x occurrences of an event cause an alarm to be triggered. If, for example, incorrect entry of passwords is to be monitored, specifying TIME-LIMIT=UNDEFINED will eventually cause the alarm to be triggered even if a user enters the password incorrectly (perhaps due to a typing error) only once per week. Alarms of this kind are clearly less effective; for this reason, long-time monitoring is better executed by evaluation of the SATLOG files.

TIME-LIMIT = *WITHIN(...)
The period within which the specified number of events must occur in order to trigger an alarm. Values must be specified for all three operands.

DAYS = <integer 0..365>
Specification of the period in days.

HOURS = <integer 0..23>
Specification of the period in hours.

MINUTES = <integer 0..59>
Specification of the period in minutes.

REPEAT= 3 / <integer 1..255>
The number of times the event must occur within the specified period in order to trigger an alarm.

TRIGGER-ACTION = *OPERATOR-MESSAGE(...)
The action to be executed when the alarm is triggered, and the expected response to this action. In this version, the only possible action is the output of a message (SAT2200) on the operator console.

WAIT-RESPONSE = *YES / *NO
Specifies whether or not the message must be acknowledged.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed Warning: user is unknown
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	64	SAT1020	Event already exists in event list
	64	SAT1022	Field already exists in field list
	64	SAT1023	Field contains duplicate values
	64	SAT1026	Specified time limit invalid
	64	SAT1027	Alarm already exists
	64	SAT1029	Event unknown
	64	SAT1030	User already exists in user list
	64	SAT1035	Value is not a multiple of 512 or too big
	64	SAT1050	Command permitted only if logging function is activated
	64	SAT1071	Alarm table is full
	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared

Notes

1. There are no predefined alarm definitions. When SAT is started for the first time, there is no parameter file and it is thus not possible to read any definitions from this file.
2. It is, however, possible to save a SAT parameter file for the next session with the aid of the /SAVE-SAT-PARAMETERS command. The next time SAT is started, definitions with the default values are then available. There are no default values for alarm definitions; if the current values are not stored in the SAT parameter file, no alarm definitions will exist for the next session.
3. Up to 32 alarm definitions can be stored.
4. If an alarm definition contains a product event for which the activation of SAT support can be controlled with /MODIFY-SAT-SUPPORT-PARAMETERS (in the current version, this is restricted to POSIX) then, if the event occurs, this alarm can only be issued if SAT support is activated for the product in question.
5. When evaluating an alarm condition with a UNIT entry, only the value resulting from multiplying the VALUE and UNIT entries together is relevant, but not how this value is reached.

Examples

The following values are considered to be equivalent since they all represent the same value of 3145728 bytes:

```
VALUE=3145728(UNIT=*BYTES)
VALUE=3072(UNIT=*KB)
VALUE=3(UNIT=*MB)
```

a. An ADD-SAT-ALARM-CONDITIONS command with the entry

   `FIELD-NAME=*FILPOS(VALUE=(3072(UNIT=*KB),3(UNIT=*MB)))`

   is therefore rejected with the following message:

   `SAT1023 FIELD 'FILPOS' CONTAINS DUPLICATE VALUES. COMMAND REJECTED`

b. An alarm condition with the following entry

   `FIELD-NAME=*FILPOS(VALUE=3072(UNIT=*KB))`

   is valid if the record to be logged contains `FILPOS=6144`. Reason: the entry in the record represents a multiple of 512 bytes (see “filpos in Table of auditable information (field names)”) and 6144*512 Bytes = 3145728 Bytes = 3072 KB.

6. Posix filenames und Kerberos names are logged by SAT without any restriction. The following SAT fields are case-sensitive in the definition of SAT alarm conditions: AUDITID, HOMEDIR, LINKNAM, NEWPATH, PATHNAM, PRINCCL, PRINCSV, SHELL, SYMBDEV. With the exception of SYMBDEV, however, these field can be specified with a maximum length of 255 bytes only. Events with longer field contents may be specified by using wildcards. In the specification of a single name (without wildcard) the same special characters are allowed as for posix filenames or Kerberos names.
7. See also the general notes on SAT commands on "Functional overview".

Example

Each incorrect attempt to log on to terminal DSN30151 under the user ID SYSPRIV is to trigger an alarm (for the purposes of this example, it is assumed that the specified terminal is mostly used by the security administrator):

```
/add-sat-alarm-conditions name=badlogon,select=*parameters( -
/        event-name=jde(result=*failure), -
/        user-identification=syspriv, -
/        field=station(value='dsn30151')),repeat=1
```

2.5.3 ADD-SAT-FILTER-CONDITIONS Define filter conditions

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The security administrator uses this command to define conditions relating to preselection for SAT logging.
The filter definition can be displayed by the /SHOW-SAT-FILTER-CONDITIONS command. It can be removed again by means of /REMOVE-SAT-FILTER-CONDITIONS.

If preselection is possible for an event, this condition is also applied in order to decide whether or not the event is to be audited.

The events forming the basis for this decision are defined as follows:

- by the event name and the result on occurrence of the event
- by the user ID of the recorded event
- by the information relating to the event

Notes

- Audit records to which no filter condition applies are recorded.
- If a single filter condition applies to an audit record, then the action specified in this condition is the action required to be taken.
- If two or more filter conditions apply simultaneously to an audit record, then the following two cases must be differentiated:

  1. If at least one of the applicable filter conditions in the TRIGGER-ACTION operand contains the specification *LOGGING(RECORDING=*YES), the audit record is recorded.
  2. Only if **all** the applicable filter conditions in the TRIGGER-ACTION operand contain the specification *LOGGING(RECORDING=*NO) will the audit record not be recorded.

ADD-SAT-FILTER-CONDITIONS

```
NAME = <name 1..8>
, SELECT = *PARAMETERS(...)
    *PARAMETERS(...)
        |  EVENT-NAME = *ALL / list-poss(50): <name 3..3>(...)
        |       <name 3..3>(...)
        |           |  RESULT = *ALL / *SUCCESS / *FAILURE
        |  , USER-IDENTIFICATION = *ALL / list-poss(50): <name 1..8>
        |  , FIELD-NAME = *ALL / list-poss(50): <name 3..7>(...)
        |       <name 3..7>(...)
        |           |  VALUE = *ALL / *MATCH(...) / *NOT-MATCH(...) / list-poss(10): <text> /
        |           |                    list-poss(10): <integer 0..2147483647>(...)
        |           |       *MATCH(PATTERN=<text>)
        |           |       *NOT-MATCH(PATTERN=<text>)
        |           |       <integer 0..2147483647>(...)
        |           |           |  UNIT = *BYTES / *KB / *MB / *GB
, TRIGGER-ACTION = *LOGGING (...)
    *LOGGING (...)
        |  RECORDING = *YES / *NO
```

NAME = <name 1..8>
Name of the filter.

SELECT = *PARAMETERS(...)
This specifies which events satisfy the filter condition.

EVENT-NAME =
Type and result of the events which satisfy the filter condition.

EVENT-NAME = *ALL
All events which can be recorded by SAT satisfy the filter condition.

EVENT-NAME = list-poss(50): <name 3..3>(...)
The explicit name of an event. This name must be taken from “Table of object-related events”.

RESULT = *ALL / *SUCCESS / *FAILURE
Specifies the result the event is to have.

USER-IDENTIFICATION =
Specifies which user IDs satisfy the filter condition.

USER-IDENTIFICATION = *ALL
All user IDs satisfy the filter condition.

USER-IDENTIFICATION = list-poss(50): <name 1..8>
Only events which concern the specified user IDs satisfy the filter condition. The user IDs do not need to exist at the time when the filter condition is defined.

FIELD-NAME =
This specifies which field of an event is to be monitored.

FIELD-NAME = *ALL
All data fields of an event are checked.

FIELD-NAME = list-poss(50): <name 3..7>(...)
Only a data field specified here is checked. A list of the possible field names can be found in “Tables of auditable information on object-related events (1)”.

VALUE = *ALL / *MATCH(...) / *NOT-MATCH(...) / list-poss(10): <text> / list-poss(10): <integer 0..2147483647>(...)
A list of the field names and the information output in these fields can be found in “ Tables of auditable information on object-related events (1) ”. <text> depends on the field being logged.

VALUE = *MATCH(...)
Specifies a pattern for the field name. The condition is valid when the comparison value fits into this pattern. Pattern specification is permitted only for field names whose values represent a character string (<c-string>, <filename>, <name>).

PATTERN = <text>
Pattern specification in the format <c-string 1..255> where, analogously to the SDF data type <c-string with-wild (n)>, parts of the character string can be replaced by wildcards.
The available wildcard characters are as follows:

*	Stands for any desired character string, including a blank string
/	Stands for precisely one character
\	Nullifies the effect of “wildcards” (* / < > : ,) actually forming part of the character string (e.g. ab*c denotes the actual character string “ab*c”)

<s_x:s_y>	Replaces a character string where the following applies: • it is at least as long as the shortest character string (s_x or s_y) • it is at most as long as the longest character string (s_x or s_y) • it falls between s_x and s_y in the alphabetical sort sequence; numbers are sorted after letters (A...Z 0...9) • s_x may also be the blank character string which appears at the beginning of the alphabetical sort sequence • s_y may also be the blank character string which stands at this position for the character string with the highest possible coding (contains only the characters X‘FF ‘) • s_x must precede s_y in the alphabetical sort sequence. If s_x is shorter than s_y, s_x will be padded with X‘00‘ • if s_y is shorter than s_x, s_y will be padded with X‘FF‘ • no wildcards may occur either in s_x or in s_y
<s1,...>	Replaces all character strings to which one of the character combinations specified by s applies. s may also be a blank character string. Any character string s may also be a range specification <s_x:s_y>

VALUE = *NOT-MATCH(...)
Specifies a pattern for the field name. The condition is valid when the comparison value does **not** fit into this pattern. Pattern specification is permitted only for field names whose values represent a character string (<c-string>, <filename>,
<name>).

PATTERN = <text>
Pattern specification as under VALUE=*MATCH.

VALUE = <integer 0..2147483647>(...)
Specifies a numerical value for the field name. This value is only allowed for fields whose value is of type <integer>.

UNIT = *BYTES / *KB / *MB / *GB
Specifies the units to be used in interpreting the value specified with the VALUE operand. This entry is only allowed for field names filpos, curlim2 and maxlim2.

The following thereby applies:

- If UNIT=*BYTES is implicitly or explicitly defined, the value must be a multiple of 512.

- The maximum value of 2^{40}-512 (=1 099 511 627 264) bytes may also not be exceeded if UNIT=*KB / *MB / *GB is specified. This results in the following maximum values, depending on the UNIT entry:

UNIT=	Maximum value for VALUE	Corresponds in bytes to
***BYTES**	2^{31}-1 = 2 147 483 647	2^{31}-1 = 2 147 483 647
***KB**	2^{30}-1 = 1 073 741 823	2^{40}-2^{10} = 1 099 511 626 752
***MB**	2^{20}-1 = 1 048 575	2^{40}-2^{20} = 1 099 510 579 200
***GB**	2^{10}-1 = 1 023	2^{40}-2^{30} = 1 098 437 885 952

TRIGGER-ACTION = *LOGGING(...)
Specifies which action is to be performed when the condition defined with the SELECT operand is satisfied.

RECORDING =
Specifies whether an event is to be recorded.

RECORDING = *YES
The event is recorded.

RECORDING = *NO
The event is not recorded, provided no other filter condition calls for recording.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed Warning: user is unknown
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	64	SAT1020	Event already exists in event list
	64	SAT1022	Field already exists in field list
	64	SAT1023	Field contains duplicate values
	64	SAT1029	Event unknown
	64	SAT1030	User already exists in user list
	64	SAT1031	Filter already exists
	64	SAT1035	Value is not a multiple of 512 or too big
	64	SAT1050	Command permitted only if logging function is activated
	64	SAT1073	Filter table is full

	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared

Notes

1. There are no predefined filter definitions. When SAT is started for the first time, there is no parameter file and it is thus not possible to read any definitions from this file.
2. It is, however, possible to save a SAT parameter file for the next session with the aid of the /SAVE-SAT-PARAMETERS command. The next time SAT is started, definitions with the default values are then available. There are no default values for filter definitions; if the current values are not stored in the SAT parameter file, no filter definitions will exist for the next session.
3. Up to 32 alarm definitions can be stored.
4. The use of a negative list of field names and the trigger action RECORDING=*YES do not generally result in a reduction in the scope of recording since an audit record generally contains fields which then require recording.
5. When evaluating a filter condition with a UNIT entry, only the value resulting from multiplying the VALUE and UNIT entries together is relevant, but not how this value is reached.

   Examples

   The following values are considered to be equivalent since they all represent the same value of 3145728 bytes:

   ```
   VALUE=3145728(UNIT=*BYTES)
   VALUE=3072(UNIT=*KB)
   VALUE=3(UNIT=*MB)
   ```

   a. An ADD-SAT-FILTER-CONDITIONS command with the entry

      `FIELD-NAME=*FILPOS(VALUE=(3072(UNIT=*KB),3(UNIT=*MB)))`

      is therefore rejected with the following message:

      `SAT1023 FIELD 'FILPOS' CONTAINS DUPLICATE VALUES. COMMAND REJECTED`

   b. A filter condition with the following entry

      `FIELD-NAME=*FILPOS(VALUE=3072(UNIT=*KB))`

      is valid if the record to be logged contains `FILPOS=6144`. Reason: the entry in the record represents a multiple of 512 bytes (see “filpos” in Table of auditable information (field names)) and 6144*512 Bytes = 3145728 Bytes = 3072 KB.
6. Posix filenames und Kerberos names are logged by SAT without any restriction. The following SAT fields are case-sensitive in the definition of SAT filter conditions: AUDITID, HOMEDIR, LINKNAM, NEWPATH, PATHNAM, PRINCCL, PRINCSV, SHELL, SYMBDEV. With the exception of SYMBDEV, however, these field can be specified with a maximum length of 255 bytes only. Events with longer field contents may be specified by using wildcards. In the specification of a single name (without wildcard) the same special characters are allowed as for posix filenames or Kerberos names.
7. See also the general notes on SAT commands in "Functional overview".

Example

1. The following accesses are to be recorded if they refer to files which are cataloged in the catalog “CAT1” and their names contain the character strings “SYS” and “ABC”: “Read protection attributes” (FRS), if successful, and “Export catalog” (CEP)

```
/add-sat-filter-conditions name=filter1,select=*parameters( -
/        event-name=(frs,cep),trigger-action=*logging(recording=*no)
/add-sat-filter-conditions name=filter2,select=*parameters( -
/        event-name=(frs(result=*success),cep),user-identification=*all,-
/        field-name=(filname(value=*match(pattern='*sys*abc')), -
/                    catid(value='cat1')))
```

2. Accesses to files having names beginning with “$TSOS.SYSLNK.” are not to be recorded.

   ```
   /add-sat-filter-conditions name=f1,select=*parameters( -
   /        event-name=*all,user-identification=*all,-
   /        field-name=filname(value=*match(pattern='*$tsos.syslnk.*')), -
   /        trigger-action=*logging(recording=*no)
   ```

   The “Delete file” event (FDD), should be recorded for all files, however:

   ```
   /add-sat-filter-conditions name=f2,select=*parameters( -
   /        event-name=fdd,user-identification=*all,field-name=*all), -
   /        trigger-action=*logging(recording=*yes)
   ```

   As regards the deletion of a file whose name begins with $TSOS.SYSLNK., both conditions are applicable. Since one of these conditions calls for recording, the corresponding audit record is recorded.

Further examples may be found under /MODIFY-SAT-FILTER-CONDITIONS.

2.5.4 CHANGE-SAT-FILE Change SATLOG file

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION, SAT-FILE-MANAGEMENT

The /CHANGE-SAT-FILE command is used by the SAT file manager to close the current SATLOG file and to open a new one during normal system operation. No data is lost during the changeover. New attributes can be specified for the new SATLOG file.

Using the functionality of this command, we can move the SATLOG file to any available pubset. The first SATLOG file created in a session is always created on the home pubset. If there is the pubset other than home pubset and default pubset in the SAT parameters, SAT waits until it receives information that the given pubset was imported, then moves the SATLOG file to it. If pubset is to be exported, SAT is informed and automatically moves the SATLOG file back to home pubset.
The security administrator can also use this command to change the logging file, but in contrast is allowed only to use the default values of the operands.

```
CHANGE-SAT-FILE

 BUFFER-LENGTH = *UNCHANGED / *STD(...)

     *STD(...)

         |    SIZE = 2 / <integer 1..16>

 ,SPACE = *UNCHANGED / *RELATIVE(...)

     *RELATIVE(...)

         |    PRIMARY-ALLOCATION = 1002 / <integer 6..50331645>

         |    ,SECONDARY-ALLOCATION = 1002 / <integer 2..32767>

 ,SUPPORT = *UNCHANGED / *PUBLIC(...)

     *PUBLIC(...)

         |    PUBSET= *HOME / <cat-id 1..4>

 ,REPEAT = *UNCHANGED / *NO / *PERIOD(...)

     *PERIOD(...)

         |    DAYS = 1 / <integer 0..10>

         |    ,HOURS = 0 / <integer 0..23>
```

BUFFER-LENGTH =
This defines the buffer size for the SATLOG file.

BUFFER-LENGTH = *UNCHANGED
The current buffer size is retained.

BUFFER-LENGTH = *STD(...)

This defines a new buffer size. The default size is 2 PAM pages. A small value for BUFFER-LENGTH increases the input/output rate and minimizes the amount of data lost in the event of a system crash.

SIZE = 2 / <integer 1..16>
The default of the new buffer size is 2 PAM pages.

SPACE =
Specifies the storage space allocation.

SPACE = *UNCHANGED
The current storage space allocation is retained.

SPACE = *RELATIVE(...)
This defines the size of the primary and the secondary allocations.

PRIMARY-ALLOCATION = 1002 / <integer 6..50331645>
Primary allocation.

SECONDARY-ALLOCATION = 1002 / <integer 2..32767>
Secondary allocation.
A large secondary allocation must be selected if a large number of events are to be audited; a small secondary allocation is sufficient if the installation is small and the default values are used.

SUPPORT =
This defines the disk storages on which the SATLOG files are to be created.

SUPPORT = *UNCHANGED

The current disk storage is retained.

SUPPORT = *PUBLIC(...)
The SATLOG files are created on public disks. The first SATLOG file created in a session is always created on a public disk.

PUBSET = *HOME

The SATLOG file is moved in the disk storage of the home pubset.

PUBSET = <cat-id 1..4>

Catalog identifier of the pubset where the SATLOG file is moved. The pubset must be imported locally for this.

REPEAT = *UNCHANGED / *NO / *PERIOD(...)
This defines whether the SATLOG file is to be changed periodically.

REPEAT = *NO
The SATLOG file is not changed periodically.

REPEAT = *PERIOD(...)
The SATLOG file is to be changed periodically.

DAYS=1 / <integer 0..10>
Specification of the interval for periodic change, in days.

HOURS = 0 / <integer 0..23>
Specification of the interval for periodic change, in hours.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed
	32	SAT0000	Unrecoverable error
	32	SAT2030	DMS error when opening file
	64	SAT2040	DMS error when creating catalog entry
	64	SAT1000	User not privileged for command
	64	SAT1025	Invalid time specified for periodic change
	128	SAT1050	Command permitted only if logging function is activated
	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared

Notes

1. Any modification remains valid until another /CHANGE-SAT-FILE command with different operands is issued or until the next shutdown. If the selected pubset is different than home pubset and default pubset then, we can save the name of this pubset as the SAT parameter using the command /SAVE-SAT-PAR SAT-FILE-ATTR=*CURRENT.
2. See also the general notes on SAT commands on "Functional overview".

Example

SAT file management wishes to

- change the SATLOG file (explicit changeover)

  ```
  /change-sat-file
  ```

- create the new SATLOG file on pubset USRP

  /change-sat-file support = *public(pubset = usrp)

- return the SATLOG file to home pubset

  /change-sat-file support = *public(pubset = *home)

2.5.5 HOLD-SAT-LOGGING Suspend SAT logging

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The security administrator can use the /HOLD-SAT-LOGGING command to suspend SAT logging and the alarm function and to close the current SATLOG file. SATCP is then in the HOLD state. Not all of the commands are available in the HOLD state (see section “SAT commands”).

HOLD-SAT-LOGGING

This command has no operands.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared
	128	SAT2010	Logging function already in HOLD state

Notes

1. While SATCP is in the NO-RESOURCE state, user jobs waiting for auditing can be resumed after entry of HOLD-SAT-LOGGING.

   If SATLOG records for the waiting jobs are still buffered in class 5 memory, it is subsequently necessary to enter the RESUME-SAT-LOGGING command in order to resume processing of these jobs.

   Until such time as processing of the waiting jobs has been resumed by the RESUME-SAT-LOGGING command, SATCP issues a message at regular intervals which indicates that there are still waiting jobs in existence.
2. See also the general notes on SAT commands on "Functional overview".

2.5.6 MODIFY-SAT-ALARM-CONDITIONS Modify alarm definitions

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The /MODIFY-SAT-ALARM-CONDITIONS command can be used to modify an existing alarm definition (/ADD-SAT-ALARM-CONDITIONS).

```
MODIFY-SAT-ALARM-CONDITIONS

NAME = <name 1..8>

, SELECT = *PARAMETERS(...)

    *PARAMETERS(...)

        |   EVENT-NAME = *UNCHANGED / *ALL / list-poss(50): <name 3..3>(...)

        |       <name 3..3>(...)

        |           |   SELECT-SWITCH = *ON (...) / *OFF

        |           |       *ON(...)

        |           |           |   RESULT = *ALL / *SUCCESS / *FAILURE

        |   , USER-IDENTIFICATION = *UNCHANGED / *ALL / list-poss(50): <name 1..8>(...)

        |       <name 1..8>(...)

        |           |   SELECT-SWITCH = *ON / *OFF
```

```
        |   , FIELD-NAME = *UNCHANGED / *ALL / list-poss(50): <name 3..7>(...)
        |       <name 3..7>(...)
        |           |   SELECT-SWITCH = *ON (...) / *OFF(...)
        |           |       *ON(...)
        |           |           |   VALUE = *ALL / *MATCH(...) / *NOT-MATCH(...) / list-poss(10): <text> /
        |           |           |                   list-poss(10): <integer 0..2147483647>(...)
        |           |           |       *MATCH(PATTERN=<text>)
        |           |           |       *NOT-MATCH(PATTERN=<text>)
        |           |           |       <integer 0..2147483647>(...)
        |           |           |           |   UNIT = *BYTES / *KB / *MB / *GB
        |           |       *OFF(...)
        |           |           |   VALUE = *ALL / *MATCH(...) / *NOT-MATCH(...) / list-poss(10): <text> /
        |           |           |                   list-poss(10): <integer 0..2147483647>(...)
        |           |           |       *MATCH(PATTERN=<text>)
        |           |           |       *NOT-MATCH(PATTERN=<text>)
        |           |           |       <integer 0..2147483647>(...)
        |           |           |           |   UNIT = *BYTES / *KB / *MB / *GB
, TIME-LIMIT = *UNCHANGED / *UNDEFINED / *WITHIN(...)
    *WITHIN(...)
        |   DAYS = <integer 0..365>
        |   , HOURS = <integer 0..23>
        |   , MINUTES = <integer 0..59>
, REPEAT = *UNCHANGED / <integer 1..255>
, TRIGGER-ACTION = *UNCHANGED / *OPERATOR-MESSAGE(...)
    *OPERATOR-MESSAGE(...)
        |   WAIT-RESPONSE = *YES / *NO
```

NAME = <name 1..8>
Name of the alarm.

SELECT = *PARAMETERS(...)
This specifies which of the existing conditions are to be modified.

EVENT-NAME = *UNCHANGED / *ALL / list-poss(50): <name 3..3>(...)
Type and result of the event(s) to be monitored.

EVENT-NAME = *ALL
All events which can be recorded by SAT are to be monitored for the alarm function.

EVENT-NAME = list-poss(50): <name 3..3>(...)
Explicit name of an event. The name of the event must be taken from "Table of objec-trelated events". If you specify POSIX events, please pay special attention to Note 6.

SELECT-SWITCH =
This specifies whether the event is to be added or removed.

SELECT-SWITCH = *ON(...)
The event and result are to be added to the alarm definition.

RESULT = *ALL / *SUCCESS / *FAILURE
This specifies the result the event is to have.

SELECT-SWITCH = *OFF
The event is to be removed from the alarm definition.

USER-IDENTIFICATION = *UNCHANGED / *ALL / list-poss(50): <name 1..8>(...)
User IDs which are to be monitored.

USER-IDENTIFICATION = *ALL
All user IDs are to be monitored.

USER-IDENTIFICATION = list-poss(50): <name 1..8>(...)
The specified user IDs are to be monitored. The user ID does not need to exist at the time when the alarm condition is defined.

SELECT-SWITCH =
User ID to be added to or deleted from the alarm definition.

SELECT-SWITCH = *ON
The user ID is to be added to the alarm definition.

SELECT-SWITCH = *OFF
The user ID is to be deleted from the alarm definition.

FIELD-NAME = *UNCHANGED / *ALL / list-poss(50): <name 3..7>(...) This specifies which field of an event is to be monitored.

FIELD-NAME = *ALL
All fields of an event are to be monitored.

FIELD-NAME = list-poss(50): <name 3..7>(...)
Only the field(s) specified here are to be monitored. A list of the possible field names can be found in "Tables of auditable information on object-related events (1)".

SELECT-SWITCH =
The information to be monitored is to be added to or deleted from the alarm definition if it corresponds to a value specified using the VALUE operand. The table of field names together with the output information can be found in “Tables of auditable information on object-related events (1)”. <text> depends on the logged data field.

SELECT-SWITCH = *ON(...)

The information to be monitored is to be added to the alarm definition.

VALUE = *ALL
All information is to be monitored.

VALUE = *MATCH(...)
Specifies a pattern for the information. The condition is valid when the comparison value fits into this pattern. Pattern specification is permitted only for field names whose values represent a character string (<c-string>, <filename>, <name>).

PATTERN = <text>
Pattern specification in the format c-string 1..255 where, analogously to the SDF data type <c-string with-wild (n)>, parts of the character string can be replaced by wildcards.
The available wildcard characters are as follows:

*	Stands for any desired character string, including a blank string
/	Stands for precisely one character
\	Nullifies the effect of “wildcards” (* / < > : ,) actually forming part of the character string (e.g. ab*c denotes the actual character string “ab*c”)

<s_x:s_y>	Replaces a character string where the following applies: • it is at least as long as the shortest character string (s_x or s_y) • it is at most as long as the longest character string (s_x or s_y) • it falls between s_x and s_y in the alphabetical sort sequence; numbers are sorted after letters (A...Z 0...9) • s_x may also be the blank character string which appears at the beginning of the alphabetical sort sequence • s_y may also be the blank character string which stands at this position for the character string with the highest possible coding (contains only the characters X‘FF‘) • s_x must precede s_y in the alphabetical sort sequence. If s_x is shorter than s_y, s_x will be padded with X‘00‘ • if s_y is shorter than s_x, s_y will be padded with X‘FF‘ • no wildcards may occur either in s_x or in s_y
<s1,...>	Replaces all character strings to which one of the character combinations specified by s applies. s may also be a blank character string. Any character string s may also be a range specification <s_x:s_y>

VALUE = *NOT-MATCH(...)
Specifies a pattern for the information. The condition is valid when the comparison value does **not** fit into this pattern. Pattern specification is permitted only for field names whose values represent a character string (<c-string>, <filename>, <name>).

PATTERN = <text>
Pattern specification as under VALUE=*MATCH.

VALUE = list-poss(10): <text>
The explicitly specified information for the event is to be added to the alarm definition. <text> depends on the field being logged. A list of the field names and the information output in these fields can be found in “Tables of auditable information on object-related events (1)”.

VALUE = list-poss(10): <integer 0..2147483647>(...)
The information specified explicitly for the field in the form of a numerical value is monitored. This entry is only allowed for field names whose value is of type <integer>.

UNIT = *BYTES / *KB / *MB / *GB
Specifies the units to be used in interpreting the value specified with the VALUE operand. This entry is only allowed for field names filpos, curlim2 and maxlim2.

The following thereby applies:

- If UNIT=*BYTES is implicitly or explicitly defined, the value must be a multiple of 512.
- The maximum value of 2^{40}-512 (=1 099 511 627 264) bytes may also not be exceeded if UNIT=*KB / *MB / *GB is specified.
  This results in the following maximum values, depending on the UNIT entry:

UNIT=	Maximum value for VALUE	Corresponds in bytes to
***BYTES**	2^{31}-1 = 2 147 483 647	2^{31}-1 = 2 147 483 647
***KB**	2^{30}-1 = 1 073 741 823	2^{40}-2^{10} = 1 099 511 626 752
***MB**	2^{20}-1 = 1 048 575	2^{40}-2^{20} = 1 099 510 579 200
***GB**	2^{10}-1 = 1 023	2^{40}-2^{30} = 1 098 437 885 952

SELECT-SWITCH = *OFF
The information to be monitored is to be deleted from the alarm definition.

VALUE = *ALL
All information is to be deleted.

VALUE = *MATCH(...)
Specifies a pattern for the information. The condition is valid when the comparison value fits into this pattern. Pattern specification is permitted only for field names whose values represent a character string (<c-string>, <filename>, <name>).

PATTERN = <text>
Pattern specification in the format c-string 1..255 where, analogously to the SDF data type <c-string with-wild (n)>, parts of the character string can be replaced by wildcards.
The available wildcard characters are as follows:

*	Stands for any desired character string, including a blank string
/	Stands for precisely one character
\	Nullifies the effect of “wildcards” (* / < > : ,) actually forming part of the character string (e.g. ab*c denotes the actual character string “ab*c”)

<s_x:s_y>	Replaces a character string where the following applies: • it is at least as long as the shortest character string (s_x or s_y) • it is at most as long as the longest character string(s_x or s_y) • it falls between s_x and s_y in the alphabetical sort sequence; numbers are sorted after letters (A...Z 0...9) • s_x may also be the blank character string which appears at the beginning of the alphabetical sort sequence • s_y may also be the blank character string which stands at this position for the character string with the highest possible coding (contains only the characters X‘FF‘) • s_x must precede s_y in the alphabetical sort sequence. If s_x isshorter than s_y, s_x will be padded with X‘00‘ • if s_y is shorter than s_x, s_y will be padded with X‘FF‘ • no wildcards may occur either in s_x or in s_y
<s1,...>	Replaces all character strings to which one of the character combinations specified by s applies. s may also be a blank character string. Any character string s may also be a range specification <s_x:s_y>

VALUE = *NOT-MATCH(...)
Specifies a pattern for the information. The condition is valid when the comparison value does **not** fit into this pattern. Pattern specification is permitted only for field names whose values represent a character string (<c-string>, <filename>, <name>).

PATTERN = <text>
Pattern specification as under VALUE=*MATCH.

VALUE = list-poss(10): <text>
The explicitly specified information for the event is to be deleted from the alarm definition. <text> depends on the field being logged. A list of the field names and the information output in these fields can be found in “Tables of auditableinformation on object-related events (1)”.

VALUE = list-poss(10): <integer 0..2147483647>(...)
The information specified explicitly for the field in the form of a numerical value is removed from the alarm definition. This entry is only allowed for field names whose value is of type <integer>.

UNIT = *BYTES / *KB / *MB / *GB
Specifies the units to be used in interpreting the value specified with the VALUE operand. This entry is only allowed for field names filpos, curlim2 and maxlim2.

The following thereby applies:

- If UNIT=*BYTES is implicitly or explicitly defined, the value must be a multiple of 512.

- The maximum value of 2^{40}-512 (=1 099 511 627 264) bytes may also not be exceeded if UNIT=*KB / *MB / *GB is specified.
  This results in the following maximum values, depending on the UNIT entry:

UNIT=	Maximum value for VALUE	Corresponds in bytes to
***BYTES**	2^{31}-1 = 2 147 483 647	2^{31}-1 = 2 147 483 647
***KB**	2^{30}-1 = 1 073 741 823	2^{40}-2^{10} = 1 099 511 626 752
***MB**	2^{20}-1 = 1 048 575	2^{40}-2^{20} = 1 099 510 579 200
***GB**	2^{10}-1 = 1 023	2^{40}-2^{30} = 1 098 437 885 952

TIME-LIMIT = *UNCHANGED / *UNDEFINED / *WITHIN(...)
The period within which x (defined with REPEAT) occurrences of an event are to trigger an alarm.

TIME-LIMIT = *UNDEFINED
The entire period of SAT logging is to be evaluated. This means that x occurrences of an event cause an alarm to be triggered. If, for example, incorrect entry of passwords is to be monitored, specifying TIME-LIMIT=*UNDEFINED will eventually cause the alarm to be triggered even if a user enters the password incorrectly (perhaps due to a typing error) only once per week. Alarms of this kind are clearly less effective; for this reason, long-time monitoring is better executed by evaluation of the SATLOG files.

TIME-LIMIT = *WITHIN(...)
The period within which the specified number of events must occur in order to trigger an alarm. Values must be specified for all three operands.

DAYS = <integer 0..365>
Specification of the period, in days.

HOURS = <integer 0..23>
Specification of the period, in hours.

MINUTES = <integer 0..59>
Specification of the period, in minutes.

REPEAT= *UNCHANGED / <integer 1..255>
The number of times the event must occur within the specified period in order to trigger an alarm.

TRIGGER-ACTION = *UNCHANGED / *OPERATOR-MESSAGE(...)
The action to be executed when the alarm is triggered, and the expected response to this action. In this version, the only possible action is the output of a message (SAT2200) on the operator console.

TRIGGER-ACTION = *OPERATOR-MESSAGE(...)
Specifies the expected response to the output of the message.

WAIT-RESPONSE = *YES / *NO
Specifies whether or not the message must be acknowledged.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed Warning: user unknown Warning: alarm not triggered Warning: more than one warning issued
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	64	SAT1020	Event already exists in event list
	64	SAT1022	Field already exists in field list
	64	SAT1023	Field contains duplicate values
	64	SAT1026	Specified time limit invalid
	64	SAT1028	Alarm unknown
	64	SAT1029	Event unknown
	64	SAT1030	User already exists in user list
	64	SAT1035	Value is not a multiple of 512 or too big
	64	SAT1050	Command permitted only if logging function is activated
	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared

Notes

1. When using patterns for values of a field no check is made as to whether any overlaps occur.
2. Identically specified patterns for a value of a field are replaced.

   Examples

   Let us assume that an alarm condition is defined as follows:

   ```
   /add-sat-alarm-conditions name=alarm1, ... -
   /        field-name=filname(value=*match('*abc*')), ...
   ```

   a. The command

   ```
   /modify-sat-alarm-conditions name=alarm1, ... -
   /  field-name=filname( -
   /       select-switch=*on(value=*not-match('*abc*'))), ...
   ```

   overwrites the comparison pattern. The effect is as if the condition had been defined in the following manner:

   ```
   /add-sat-alarm-conditions name=alarm1, ... -
   /        field-name=filname(value=*not-match('*abc*')), ...
   ```

b. Either specifying SELECT-SWITCH=*OFF(VALUE=*MATCH('*ABC*')) or specifying SELECT-SWITCH=*OFF(VALUE=*NOT-MATCH('*ABC*')) removes *MATCH('*ABC*') from the list of values.

3. The specification of a fixed value has no influence on a pattern specification.

   For example, a /MODIFY-SAT-ALARM-CONDITIONS command with the specification VALUE='XABCY' has no effect on an alarm condition which was defined using VALUE=*MATCH('*ABC*')). The value 'XABCY' is already present in the pattern specification '*ABC*' and the condition VALUE='XABCY' is therefore automatically fulfilled if *MATCH='*ABC*' is fulfilled.

   However, the specification VALUE='XABCY' does have an effect on an alarm condition defined with VALUE=*NOT-MATCH('*ABC*')). In this case, the condition applies to all the values which do not match the pattern '*ABC*' as well as to the value 'XABCY'.

4. SELECT-SWITCH=*OFF removes the specified objects from a list defined with SELECT-SWITCH=*ON or a corresponding /ADD-SAT-ALARM-CONDITIONS command. If *ALL is in effect, the object is included in a negative list.

   The specifications for the SELECT-SWITCH operand (in all cases) are only taken into consideration if they result in the creation of conditions. If, for example, USER-ID=*ALL was defined with the /ADD-SAT-ALARM-CONDITIONS command for an alarm, then specifying USER-ID=HUGO(SELECT-SWITCH=*ON) in the /MODIFY-SAT-ALARM-CONDITIONS command has no effect. Specifying USER-ID=HUGO(SELECT-SWITCH=*OFF) causes these fields to be entered in a negative list.

5. If a pattern is in effect for a field value, it is not possible to extract any subset from the pattern by means of SELECT-SWITCH= *OFF(VALUE=value): If, for example, an alarm condition was defined with SELECT-SWITCH=*ON(VALUE=*MATCH('*ABC*')) or a corresponding /ADD-SAT-ALARM-CONDITIONS command, a /MODIFY-SAT-ALARM-CONDITIONS command specifying SELECT-SWITCH=*OFF(VALUE= 'SYSABC') has no effect.

   Example

   Let us assume that an alarm condition is defined as follows:

   ```
   /add-sat-alarm-conditions name=alarm1, -
   /        field-name=filname(value=*match('*abc*')), ...
   ```

   The following command has no effect:

   ```
   /modify-sat-alarm-conditions name=alarm1, ... -
   /  field-name=filname( -
   /       select-switch=*off(value=:cati:$tsos.sysabc))
   ```

6. If an alarm definition contains a product event for which the activation of SAT support can be controlled with /MODIFY-SAT-SUPPORT-PARAMETERS (in the current version, this is restricted to POSIX) then, if the event occurs, this alarm can only be issued if SAT support is activated for the product in question.

7. When evaluating an alarm condition with a UNIT entry, only the value resulting from multiplying the VALUE and UNIT entries together is relevant, but not how this value is reached.

   Examples

   The following values are considered to be equivalent since they all represent the same value of 3145728 bytes:

   ```
   VALUE=3145728(UNIT=*BYTES)
   VALUE=3072(UNIT=*KB)
   VALUE=3(UNIT=*MB)
   ```

a. A MODIFY-SAT-ALARM-CONDITIONS command with the following entry

   ```
   FIELD-NAME=*FILPOS(SELECT-SWITCH=*ON(
                  VALUE=(3072(UNIT=*KB),3(UNIT=*MB))))
   ```

   is therefore rejected with the following message:

   ```
   SAT1023 FIELD 'FILPOS' CONTAINS DUPLICATE VALUES. COMMAND REJECTED
   ```

b. An alarm condition that was set with the entry `VALUE=3145728(UNIT=*BYTES)` in an ADD-SAT-ALARM-CONDITIONS command, can be removed from the alarm table with the entry `VALUE=3(UNIT=*MB)` in a MODIFY-SAT-ALARM-CONDITIONS command.

c. An alarm condition with the entry

   ```
   FIELD-NAME=*FILPOS(SELECT-SWITCH=*ON(VALUE=3072(UNIT=*KB)))
   ```

   is valid if the record to be logged contains `FILPOS=6144`. Reason: the entry in the record represents a multiple of 512 bytes (see “filpos” in Table of auditable information (field names)) and 6144*512 Bytes = 3145728 Bytes = 3072 KB.

8. Posix filenames und Kerberos names are logged by SAT without any restriction. The following SAT fields are case-sensitive in the definition of SAT alarm conditions: AUDITID, HOMEDIR, LINKNAM, NEWPATH, PATHNAM, PRINCCL, PRINCSV, SHELL, SYMBDEV. With the exception of SYMBDEV, however, these field can be specified with a maximum length of 255 bytes only. Events with longer field contents may be specified by using wildcards. In the specification of a single name (without wildcard) the same special characters are allowed as for posix filenames or Kerberos names.
9. See also the general comments on SAT commands on "Functional overview".

Example

In the example for the /ADD-SAT-ALARM-CONDITIONS command, an alarm with the name badlogon was defined. This alarm is triggered each time there is a failed attempt to log on at terminal DSN30151 under the user ID SYSPRIV:

```
/add-sat-alarm-conditions name=badlogon,select=*parameters( -
/                event-name=jde(result=*failure), -
/                user-identification=syspriv, -
/                field=station(value='dsn30151')),repeat=1
```

This alarm is now to be modified in such a way that any failed attempt to log on under the user ID SYSPRIV causes an alarm irrespective of the terminal at which it is performed. The alarm definition is modified as follows:

```
/modify-sat-alarm-conditions name=badlogon,select=*parameters( -
/ field-name=station(select-switch=*on(value=*all)))
```

2.5.7 MODIFY-SAT-FILTER-CONDITIONS Modify filter definitions

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The /MODIFY-SAT-FILTER-CONDITIONS command can be used to modify an existing filter definition (/ADD-FILTER-CONDITIONS).

MODIFY**-SAT-FILTER-COND**ITIONS

```
NAME = <name 1..8>

, SELECT = *PARAMETERS(...)

    *PARAMETERS(...)

        |   EVENT-NAME = *UNCHANGED / *ALL / list-poss(50): <name 3..3>(...)

        |         <name 3..3>(...)

        |               |     SELECT-SWITCH = *ON (...) / *OFF

        |               |           *ON(...)

        |               |                 |     RESULT = *ALL / *SUCCESS / *FAILURE

        |   , USER-IDENTIFICATION = *UNCHANGED / *ALL / list-poss(50): <name 1..8>(...)

        |         <name 1..8>(...)

        |               |     SELECT-SWITCH = *ON / *OFF
```

```
      |   , FIELD-NAME = *UNCHANGED / *ALL / list-poss(50): <name 3..7>(...)
      |       <name 3..7>(...)
      |           |   SELECT-SWITCH = *ON (...) / *OFF(...)
      |           |       *ON(...)
      |           |           |   VALUE = *ALL / *MATCH(...) / *NOT-MATCH(...) / list-poss(10): <text> /
      |           |           |                       list-poss(10): <integer 0..2147483647>(...)
      |           |           |           *MATCH(PATTERN=<text>)
      |           |           |           *NOT-MATCH(PATTERN=<text>)
      |           |           |           <integer 0..2147483647>(...)
      |           |           |               |   UNIT = *BYTES / *KB / *MB / *GB
      |           |       *OFF(...)
      |           |           |   VALUE = *ALL / *MATCH(...) / *NOT-MATCH(...) / list-poss(10): <text> /
      |           |           |                       list-poss(10): <integer 0..2147483647>(...)
      |           |           |           *MATCH(PATTERN=<text>)
      |           |           |           *NOT-MATCH(PATTERN=<text>)
      |           |           |           <integer 0..2147483647>(...)
      |           |           |               |   UNIT = *BYTES / *KB / *MB / *GB
, TRIGGER-ACTION = *UNCHANGED / *LOGGING(...)
    *LOGGING(...)
        |   RECORDING = *YES / *NO
```

NAME = <name 1..8>
Name of the filter.

SELECT = *PARAMETERS(...)
This specifies which of the existing conditions are to be modified.

EVENT-NAME = *UNCHANGED / *ALL / list-poss(50): <name 3..3>(...)
Type and result of the events which satisfy the filter condition.

EVENT-NAME = *ALL
All events which can be recorded by SAT satisfy the filter condition.

EVENT-NAME = list-poss(50): <name 3..3>(...)
Explicit name of an event. The name of the event must be taken from “Table of object related events”.

SELECT-SWITCH =
This specifies whether the event is to be added or removed.

SELECT-SWITCH = *ON(...)
The event and result are to be added to the filter definition.

RESULT = *ALL / *SUCCESS / *FAILURE This specifies the result the event is to have.

SELECT-SWITCH = *OFF
The event is to be removed from the alarm definition.

USER-IDENTIFICATION = *UNCHANGED / *ALL / list-poss(50): <name 1..8>(...)
Specifies which user IDs satisfy the filter condition.

USER-IDENTIFICATION = *ALL
All user IDs satisfy the filter condition.

USER-IDENTIFICATION = list-poss(50): <name 1..8>(...)
Only events which concern the specified user IDs satisfy the filter condition. The user IDs do not need to exist at the time when the alarm condition is defined.

SELECT-SWITCH =
User ID to be added to or deleted from the filter definition.

SELECT-SWITCH = *ON
The user ID is to be added to the filter definition.

SELECT-SWITCH = *OFF
The user ID is to be deleted from the filter definition.

FIELD-NAME = *UNCHANGED / *ALL / list-poss(50): <name 3..7>(...)
Specifies which data field of an event is to be checked. The table with the possible field names can be found in the “Tables of auditable information on object-related events (1)”.

FIELD-NAME = *ALL
All data fields of an event satisfy the filter condition.

FIELD-NAME = list-poss(50): <name 3..7>(...)
A data field is specified.

SELECT-SWITCH =
Events are added to or removed from the definition when the associated information has a value defined by means of the VALUE operand. The table of field names and the information output there may be found in the “Tables of auditable informationobject-related events (1)”. <text> depends on the logged data field.

SELECT-SWITCH = *ON(...)
Adds information requiring checking to the filter definition.

VALUE = *ALL
All information satisfies the filter condition.

VALUE = *MATCH (...)
Specifies a pattern for the information. The condition is valid when the comparison value fits into this pattern. Pattern specification is permitted only for field names whose values represent a character string (<c-string>, <filename>, <name>).

PATTERN = <text>
Pattern specification in the format c-string 1..255 where, analogously to the SDF data type <c-string with-wild (n)>, parts of the character string can be replaced by wildcards.
The available wildcard characters are as follows:

*	Stands for any desired character string, including a blank string
/	Stands for precisely one character
\	Nullifies the effect of “wildcards” (* / < > : ,) actually forming part of the character string (e.g. ab*c denotes the actual character string “ab*c”)
<s_x:s_y>	Replaces a character string where the following applies: • it is at least as long as the shortest character string (s_x or s_y) • it is at most as long as the longest character string (s_x or s_y) • it falls between s_x and s_y in the alphabetical sort sequence; numbers are sorted after letters (A...Z 0...9) • s_x may also be the blank character string which appears at the beginning of the alphabetical sort sequence • s_y may also be the blank character string which stands at this position for the character string with the highest possible coding (contains only the characters X‘FF‘) • s_x must precede s_y in the alphabetical sort sequence. If s_x is shorter than s_y, s_x will be padded with X‘00‘ • if s_y is shorter than s_x, s_y will be padded with X‘FF‘ • no wildcards may occur either in s_x or in s_y
<s1,...>	Replaces all character strings to which one of the character combinations specified by s applies. s may also be a blank character string. Any character string s may also be a range specification <s_x :s_y >

VALUE = *NOT-MATCH(...)
Specifies a pattern for the information. The condition is valid when the comparison value does **not** fit into this pattern. Pattern specification is permitted only for field names whose values represent a character string (<c-string>, <filename>, <name>).

PATTERN=<text>
Pattern specification as under VALUE=*MATCH.

VALUE = list-poss(10):<text>
The information specified explicitly for the field satisfies the filter condition.

VALUE = list-poss(10): <integer 0..2147483647>(...)
The information specified explicitly for the field in the form of a numerical value satisfies the filter condition. This entry is only allowed for field names whose value is of type <integer>.

UNIT = *BYTES / *KB / *MB / *GB
Specifies the units to be used in interpreting the value specified with the VALUE operand. This entry is only allowed for field names filpos, curlim2 and maxlim2.

The following thereby applies:

- If UNIT=*BYTES is implicitly or explicitly defined, the value must be a multiple of 512.
- The maximum value of 2^{40}-512 (=1 099 511 627 264) bytes may also not be exceeded if UNIT=*KB / *MB / *GB is specified. This results in the following maximum values, depending on the UNIT entry:

UNIT=	Maximum value for VALUE	Corresponds in bytes to
***BYTES**	2^{31}-1 = 2 147 483 647	2^{31}-1 = 2 147 483 647
***KB**	2^{30}-1 = 1 073 741 823	2^{40}-2^{10} = 1 099 511 626 752
***MB**	2^{20}-1 = 1 048 575	2^{40}-2^{20} = 1 099 510 579 200
***GB**	2^{10}-1 = 1 023	2^{40}-2^{30} = 1 098 437 885 952

SELECT-SWITCH = *OFF(...)
Removes events from the filter definition.

VALUE = *ALL
All information is removed from the filter definition.

VALUE = *MATCH(...)
Specifies a pattern for the information. The condition is valid when the comparison value fits into this pattern. Pattern specification is permitted only for field names whose values represent a character string (<c-string>, <filename>, <name>).

PATTERN = <text>
Pattern specification in the format c-string 1..255 where, analogously to the SDF data type <c-string with-wild (n)>, parts of the character string can be replaced by wildcards.
The available wildcard characters are as follows:

*	Stands for any desired character string, including a blank string
/	Stands for precisely one character
\	Nullifies the effect of “wildcards” (* / < > : ,) actually forming part of the character string (e.g. ab*c denotes the actual character string “ab*c”)

<s_x:s_y>	Replaces a character string where the following applies: • it is at least as long as the shortest character string (s_x or s_y) • it is at most as long as the longest character string (s_x or s_y) • it falls between s_x and s_y in the alphabetical sort sequence; numbers are sorted after letters (A...Z 0...9) • s_x may also be the blank character string which appears at the beginning of the alphabetical sort sequence • s_y may also be the blank character string which stands at this position for the character string with the highest possible coding (contains only the characters X‘FF‘) • s_x must precede s_y in the alphabetical sort sequence. If s_x is • shorter than s_y, s_x will be padded with X‘00‘ • if s_y is shorter than s_x, s_y will be padded with X‘FF‘ • no wildcards may occur either in s_x or in s_y
<s1,...>	Replaces all character strings to which one of the character combinations specified by s applies. s may also be a blank character string. Any character string s may also be a range specification <s_x :s_y >

VALUE = *NOT-MATCH(...)
Specifies a pattern for the information. The condition is valid when the comparison value does **not** fit into this pattern. Pattern specification is permitted only for field names whose values represent a character string (<c-string>, <filename>, <name>).

PATTERN = <text>
Pattern specification as under VALUE=*MATCH.

VALUE = list-poss(10): <text>
The explicitly specified information for the field is removed from the filter definition.

VALUE = list-poss(10): <integer 0..2147483647>(...)
The information specified explicitly for the field in the form of a numerical value is removed from the filter definition. This entry is only allowed for field names whose value is of type <integer>.

UNIT = *BYTES / *KB / *MB / *GB
Specifies the units to be used in interpreting the value specified with the VALUE operand. This entry is only allowed for field names filpos, curlim2 and maxlim2.

The following thereby applies:

- If UNIT=*BYTES is implicitly or explicitly defined, the value must be a multiple of 512.

- The maximum value of 2^{40}-512 (=1 099 511 627 264) bytes may also not be exceeded if UNIT=*KB / *MB / *GB is specified. This results in the following maximum values, depending on the UNIT entry:

UNIT=	Maximum value for VALUE	Corresponds in bytes to
***BYTES**	2^{31}-1 = 2 147 483 647	2^{31}-1 = 2 147 483 647
***KB**	2^{30}-1 = 1 073 741 823	2^{40}-2^{10} = 1 099 511 626 752
***MB**	2^{20}-1 = 1 048 575	2^{40}-2^{20} = 1 099 510 579 200
***GB**	2^{10}-1 = 1 023	2^{40}-2^{30} = 1 098 437 885 952

TRIGGER-ACTION = *UNCHANGED / *LOGGING(...)
Specifies which action is to be performed when the condition defined with the SELECT operand is satisfied.

TRIGGER-ACTION = *LOGGING(...)
Specifies whether an event is to be recorded.

RECORDING = *YES
The event is recorded.

RECORDING = *NO
The event is not recorded, provided no other filter condition calls for recording.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed Warning: user is unknown Warning: filter has no effect
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	64	SAT1020	Event already exists in event list
	64	SAT1022	Field already exists in field list
	64	SAT1023	Field contains duplicate values
	64	SAT1029	Event unknown
	64	SAT1030	User already exists in user list
	64	SAT1031	Filter already exists
	64	SAT1035	Value is not a multiple of 512 or too big
	64	SAT1050	Command permitted only if logging function is activated

	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared

Notes

1. When using patterns for values of a field no check is made as to whether any overlaps occur.
2. Identically specified patterns for a value of a field are replaced.

   Examples

   Let us assume that a filter condition is defined as follows:

   ```
   /add-sat-filter-conditions name=filter1, ... -
   /        field-name=filname(value=*match('*abc*')), ...
   ```

   a. The command

      ```
      /modify-sat-filter-conditions name=filter1, ... -
      /  field-name=filname( -
      /       select-switch=*on(value=*not-match('*abc*'))), ...
      ```

      overwrites the comparison pattern. The effect is as if the condition had been defined in the following manner:

      ```
      /add-sat-filter-conditions name=filter1, ... -
      /        field-name=filname(value=*not-match('*abc*')), ...
      ```

   b. Either specifying SELECT-SWITCH=*OFF(VALUE=*MATCH('*ABC*')) or specifying SELECT-SWITCH=*OFF(VALUE=*NOT-MATCH('*ABC*')) removes *MATCH('*ABC*') from the list of values.
3. The specification of a fixed value has no influence on a pattern specification.

   For example, a /MODIFY-SAT-FILTER-CONDITIONS command with the specification VALUE='XABCY' has no effect on a filter condition which was defined using VALUE=*MATCH('*ABC*')). The value 'XABCY' is already present in the pattern specification '*ABC*' and the condition VALUE='XABCY' is therefore automatically fulfilled if *MATCH='*ABC*' is fulfilled.

   However, the specification VALUE='XABCY' does have an effect on a filter condition defined with VALUE=*NOT-MATCH('*ABC*')). In this case, the condition applies to all the values which do not match the pattern '*ABC*' as well as to the value 'XABCY'.
4. SELECT-SWITCH=*OFF removes the specified objects from a list defined with SELECT-SWITCH=*ON or a corresponding /ADD-SAT-FILTER-CONDITIONS command. If *ALL is in effect, the object is included in a negative list.

   The specifications for the SELECT-SWITCH operand (in all cases) are only taken into consideration if they result in the creation of conditions. If, for example, USER-ID=*ALL was defined with the /ADD-SAT-FILTER-CONDITIONS command for a filter, then

   specifying USER-ID=HUGO(SELECT-SWITCH=*ON) in the /MODIFY-SAT-FILTER-CONDITIONS command has no effect. Specifying USER-ID=HUGO(SELECT-SWITCH=*OFF) causes these fields to be entered in a negative list.
5. If a pattern is in effect for a field value, it is not possible to extract any subset from the pattern by means of SELECT-SWITCH= *OFF(VALUE=value): If, for example, a filter condition was defined with SELECT-SWITCH=*ON(VALUE=*MATCH('*ABC*')) or a corresponding /ADD-SAT-FILTER-CONDITIONS command, a /MODIFY-SAT-FILTER-CONDITIONS command SELECT-SWITCH=*OFF(VALUE= 'SYSABC') specified has no effect. The desired effect can, however, be achieved through the definition of a second filter condition:

Example

Let us assume that a filter condition is defined as follows:

```
/add-sat-filter-conditions name=filter1, -
/        field-name=filname(value=*match('*abc*')), -
/        trigger-action=*logging(recording=*no), ...
```

a. The following command has no effect:

```
/modify-sat-filter-conditions name=filter1, ... -
/  field-name=filname( -
/       select-switch=*off(value=:cati:$tsos.sysabc))
```

b. The definition of a second filter condition

```
/add-sat-filter-conditions name=filter2, -
/  field-name=filname(value=:cati:$tsos.sysabc), ...
/        trigger-action=*logging(recording=*yes)
```

has the following effect:

Both these filter conditions are applicable to audit records which concern the file :CATI:$TSOS.SYSABC. Since one of the two conditions (FILTER2) calls for recording, the records are recorded. Audit records which concern other files whose names contain “ABC” are not recorded. Only the condition FILTER1 applies to them, and this excludes recording.

6. When evaluating a filter condition with a UNIT entry, only the value resulting from multiplying the VALUE and UNIT entries together is relevant, but not how this value is reached.

Examples

The following values are considered to be equivalent since they all represent the same value of 3145728 bytes:

```
VALUE=3145728(UNIT=*BYTES)
VALUE=3072(UNIT=*KB)
VALUE=3(UNIT=*MB)
```

a. A MODIFY-SAT-FILTER-CONDITIONS command with the entry

`FIELD-NAME=*FILPOS(SELECT-SWITCH=*ON(`

`VALUE=(3072(UNIT=*KB),3(UNIT=*MB))))` is therefore rejected with the following message:

`SAT1023 FIELD 'FILPOS' CONTAINS DUPLICATE VALUES. COMMAND REJECTED`

b. A filter condition that was set with the entry `VALUE=3145728(UNIT=*BYTES)` in an ADD-SAT-FILTER-CONDITIONS command, can be removed from the filter table with the entry `VALUE=3(UNIT=*MB)` in a MODIFY-SAT-FILTER-CONDITIONS command.

c. A filter condition with the following entry

`FIELD-NAME=*FILPOS(SELECT-SWITCH=*ON(VALUE=3072(UNIT=*KB)))`

is valid if the record to be logged contains `FILPOS=6144`. Reason: the entry in the record represents a multiple of 512 bytes (see “filpos” (Table of auditable information (field names))) and 6144*512 Bytes = 3145728 Bytes = 3072 KB.

7. Posix filenames und Kerberos names are logged by SAT without any restriction. The following SAT fields are case-sensitive in the definition of SAT filter conditions: AUDITID, HOMEDIR, LINKNAM, NEWPATH, PATHNAM, PRINCCL, PRINCSV, SHELL, SYMBDEV. With the exception of SYMBDEV, however, these field can be specified with a maximum length of 255 bytes only. Events with longer field contents may be specified by using wildcards. In the specification of a single name (without wildcard) the same special characters are allowed as for posix filenames or Kerberos names.
8. See also the general notes on SAT commands on "Functional overview".

Examples

1. Accesses to the files :A:$TSOS.SYSABC and :B:$SYS.SYSXXX are to be recorded only when they are effected by the users PAUL and HUGO. Two commands are needed in order to define the requisite filter:

   One filter condition must first be defined which serves to exclude from recording all accesses to the two files.

   ```
   /add-sat-filter-conditions name=filter1,select=*parameters( -
   /        event-name=*all,user-identification=*all, -
   /        field-name=filname(value=(:a:$tsos.sysabc,:b:$sys.sysxxx))), -
   /        trigger-action=*logging(*recording = *no)
   ```

   Then the filter condition has to be modified in such a way that it does not apply to the users PAUL and HUGO whose accesses will consequently be recorded.

   ```
   /modify-sat-filter-conditions name=filter1, select=*parameters( -
   /        user-id=(paul(select-switch=*off),hugo(select-switch=*off)))
   ```

2. Accesses to files are only to be recorded if the character string “SYS” or “ABC” occurs in the file name. In addition, accesses to the file :A:$TSOS.SRMLNK are to be recorded.

   The following condition excludes from recording accesses to those files whose name does not contain “SYS”:

   ```
   /add-sat-filter-conditions name=f1,select=parameters( -
   /        event-name=*all,user-identification= *all, -
   /        field-name=filname(value=*not-match(pattern='*sys*'))), -
   /        trigger-action=*logging(recording=*no)
   ```

   This would mean that only accesses to files whose name contained the character string “SYS” would be recorded.

   A second condition implements the recording requirement for the file :A:$TSOS.SRMLNK.

   ```
   /add-sat-filter-conditions name=f2, select=parameters( -
   /        field=filname(value=:a:$tsos.srmlnk)), -
   /        trigger-action = *logging (recording = *yes)
   ```

   This condition is modified in such a way that it also applies to files whose name contains “ABC”:

   ```
   /modify-sat-filter-conditions name=f2, select=parameters( -
   /        field-name=filname(select-switch=*on(value=*match('*abc*'))))
   ```

   Both filter conditions are applicable to files whose name does not contain “SYS” but does contain “ABC”. Since recording is required in one of these conditions, the access is recorded.

2.5.8 MODIFY-SAT-PRESELECTION Modify SAT preselection value

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The /MODIFY-SAT-PRESELECTION command can be used by the security administrator to specify modifications to the following:

1. the default selection values which determine the events to be logged by SATCP; actual logging is dependent on the result of the operation, the event type and the user ID
2. the selection rule (see “Selection of events to be logged” above)
3. the authorization to invoke the system exit; the exit is not activated unless the exit routine has been loaded by system administration.
4. the recording scope which serves to specify whether *EXTENDED fields are recorded. *EXTENDED fields are marked in the “Tables of auditable information on object-related events (1)” by means of an “E”.
5. the default value for the audit setting (“user audit default”) for newly created user IDs.

Irrespective of the /MODIFY-SAT-PRESELECTION, the selection of files and library members may also be affected by modifications to the audit entry in the catalog.

MODIFY**-SAT-PRESEL**ECTION

EVENT-AUDITING = ***UNCHA**NGED / list-poss(50): <name 3..3>(...)

<name 3..3>(...)

| AUDIT-SWITCH = ***ON** (...) / ***OFF**

| ***ON** (...)

| | RESULT = ***ALL** / ***SUCC**ESS / ***FAIL**URE

, USER-AUDITING = ***UNCHA**NGED / ***ALL-SWITCH**ABLE(...) / ***DEFAULT**(...) /

list-poss(50): <name 1..8>(...)

***ALL-SWITCH**ABLE(...)

| AUDIT-SWITCH = ***ON** / ***OFF**

***DEFAULT**(...)

| NEW-USER = ***ON** / ***OFF**

<name 1..8>(...)

| AUDIT-SWITCH = ***ON** / ***OFF**

, PRESELECTION-RULE = ***UNCHA**NGED / ***INDEPEND**ENT / ***FILES-BY-EVENTS**

, EXIT = ***UNCHA**NGED / ***Y**ES / ***NO**

, LOGGING-QUANTITY = ***UNCHA**NGED / ***STD** / ***EXTENDED**

EVENT-AUDITING =
This defines the events for which auditing is to be activated or deactivated.

EVENT-AUDITING = *UNCHANGED
The current selection of events to be logged is retained.

EVENT-AUDITING = list-poss(50): <name 3..3>(...)
This specifies the event for which auditing is to be activated or deactivated, using the3-character event name, e.g. FCD, FRD,... (see “Table of object-related events”). If you specify POSIX events, please pay special attention to Note 4.

AUDIT-SWITCH =
This defines which events are to be audited.

AUDIT-SWITCH = *ON(...)
The specified event is selected for auditing.

RESULT =
This defines the circumstances under which the event is to be logged:

RESULT = *ALL
The event is always to be logged.

RESULT = *SUCCESS
The event is to be logged if the operation has been successful.

RESULT = *FAILURE
The event is to be logged if the operation has not been successful.

AUDIT-SWITCH = *OFF
The specified event is not selected for auditing.

USER-AUDITING =
This serves to specify the user IDs for which the SAT preselection is to be modified. The new selection for the auditing of a user ID is entered in the user catalog and takes effect immediately.

USER-AUDITING = *UNCHANGED
The current selection of user IDs subject to auditing is retained.

USER-AUDITING = *ALL-SWITCHABLE(...)
Defines the events which are to be logged for all switchable user IDs. Switchable user IDs are all user IDs apart from the security administrator’s ID, the user ID SYSAUDIT and user IDs possessing the SAT file management privilege.

AUDIT-SWITCH = *ON / *OFF
This defines which events are to be logged.

AUDIT-SWITCH = *ON
All events triggered by a switchable user ID are to be logged.

AUDIT-SWITCH = *OFF
Events triggered by a switchable user ID will only be logged if they have been selected using the EVENT-AUDITING operand and/or affect a selected file object (dependent on the logic rule defined with the PRESELECTION-RULE operand).

USER-AUDITING = *DEFAULT(...)
Specifies the default value for the audit setting for newly created user IDs. Newly created user IDs are all user IDs which are created after execution of the current /MODIFY-SAT-PRESELECTION command.

NEW-USER = *ON / *OFF
Defines which events are to be logged.

NEW-USER = *ON
All events triggered by a newly created user ID are to be logged.

NEW-USER = *OFF
Events triggered by a newly created user ID will only be logged if they have been selected using the EVENT-AUDITING operand and/or affect a selected file object (dependent on the logic rule defined with the PRESELECTION-RULE operand).

USER-AUDITING = list-poss(50): <name 1..8>(...)
Defines for each user ID which events are to be logged.

AUDIT-SWITCH = *ON / *OFF
This defines which events are to be logged.

> For systems with BS2000 OSD/BC > V11.0:
> With this specication logging can also switched on for non-switchable user IDs, if it was switched off due to an error.

AUDIT-SWITCH = *ON
All events triggered by the respective user IDs are to be logged.

AUDIT-SWITCH = *OFF
Events triggered by the respective user ID will only be logged if they have been selected using the EVENT-AUDITING operand and/or affect a selected file object (dependent on the logic rule defined with the PRESELECTION-RULE operand).

PRESELECTION-RULE =
This defines the logic rule governing selection.

PRESELECTION-RULE = *UNCHANGED
The current selection rule is retained.

PRESELECTION-RULE = *INDEPENDENT
This forces compulsory logging of an event if either the event or the subject (user ID) or the file object (file, library, ACL) has been selected and is affected by the event. This is equivalent to ORing as follows:

subject OR event OR file object

The INDEPENDENT selection rule causes an event to be logged when the object or subject has been selected even if the event itself has not been selected. A user ID may also be logged because of certain selected events or objects (see section “Selection procedure”) even though it is **not** selected itself.

PRESELECTION-RULE = *FILES-BY-EVENTS
This rule always results in auditing provided the subject has been selected. If the subject has not been selected, no auditing takes place unless both event and file object have been selected and the event result matches their audit attributes. If the event is not a file object event, the INDEPENDENT rule applies (see section “Selection procedure”).

The logic rule for *FILES-BY-EVENTS is as follows:

subject OR (event AND file object)

EXIT = *UNCHANGED / *YES / *NO
This defines whether system exit 110 (writing of SAT data) may be invoked.

LOGGING-QUANTITY = *UNCHANGED / *STD / *EXTENDED
Determines whether *EXTENDED fields are included in the SATLOG file.

LOGGING-QUANTITY = *STD
*EXTENDED fields are not included in the SATLOG file.

LOGGING-QUANTITY = *EXTENDED
*EXTENDED fields are included in the SATLOG file.

Note

The specification of *EXTENDED is also required if *EXTENDED fields are to be evaluated by a SAT exit routine.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed Warning: unknown event Warning: event not switchable Warning: user ID unknown Warning: user ID not switchable
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	64	SAT1010	Event already exists in event list
	64	SAT1020	User already exists in user list
	64	SAT1030	Command permitted only if logging function is activated
	128	SAT1050	Another command is currently being processed
	128	SAT1080	Exchange being prepared

Notes

1. The selection settings for SAT when first used or without individual changes having been made are as follows:

User ID:	For existing user IDs, the selection settings correspond to the entries in the user catalog. In the case of newly created user IDs, all the events are logged.
Event:	default setting of security-relevant events (see “Table ofobject-related events”)
File object:	in accordance with the entries in the file catalog
Logic rule:	INDEPENDENT rule
Exit activation:	system exit 110 not active
Recording scope:	*EXTENDED fields are not recorded

2. An error message is returned if one or more of the specified user IDs do not exist in the user catalog. The command is executed for those user IDs present in the user catalog. The same rule applies to unknown event types.
3. By default, AUDIT-SWITCH=ON is set for any new user ID created with ADD-USER. If user IDs are taken over from a previous version of BS2000/OSD-BC, the user IDs retain the previous settings.
4. If an event belongs to a product for which the activation of SAT support can be controlled with /MODIFY-SAT-SUPPORT-PARAMETERS (in the current version, this is restricted to POSIX), then any setting for this event made with /MODIFY-SAT-PRESELECTION is always accepted. However, if the event occurs, this setting is only effective if SAT support is activated for the product in question.
5. See also the general notes on SAT commands on "Functional overview".

Examples

1. The security administrator wishes to:
   - have the event types READ-DATA and DELETE-DATA logged in any case
   - have RENAME FILE (DMS) logged in the event of FAILURE
   - subject the user IDs HUGO and BILL to auditing
   - exempt the user ID JAMES from auditing
   - apply the FILES-BY-EVENTS selection rule

   To this end, the security administrator first has to look up the event names for the operations ’read file’ (=FRD), ’ delete file’ (=FDD) and ’rename file’ (=FRN) under FILE in “Table of object-related events”.

   Then the security administrator must issue the following command:

```
/modify-sat-preselection -
/       event-auditing=(frd(audit-switch=*on(result=*all)), -
/                       fdd(audit-switch=*on(result=*all)), -
/                       frn(audit-switch=*on(result=*failure))), -
/       user-auditing=(hugo(audit-switch=*on), -
/                      bill(audit-switch=*on), -
/                      james(audit-switch=*off)), -
/       preselection-rule=*files-by-events
```

2. The security administrator wants to activate default system logging for all user IDs, i.e. the default setting for events' audit attribute (see "Table of object-related events"). This setting is also to apply for user IDs which are to be created in the future. To do this, two commands are necessary. The first defines audit logging for already existing user IDs. The second command applies to newly created user IDs:

```
/modify-sat-preselection -
/       user-auditing=*all-switchable(audit-switch=*off)
/modify-sat-preselection -
/       user-auditing=*default(new-user=*off)
```

2.5.9 MODIFY-SAT-SUPPORT-PARAMETERS Product-specific activation/deactivation of logging and alarms

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The security administrator can use the /MODIFY-SAT-SUPPORT-PARAMETERS command to activate or deactivate SAT logging and SAT alarms for specific products.

“Activate” means that “normal” SAT logging is performed for all events relating to the product in question, while taking account of the preselection, filters and alarms set for these events.

“Deactivate” means that neither SAT logging nor a check for triggered alarms are performed for events relating to these products. This applies irrespectively of the preselection settings for these events or ofthe alarms that are defined.

Consequently, if an event is to be logged, the setting made with /MODIFY-SAT-PRESELECTION is only effective if the event does not relate to a product for which SAT support has been deactivated by means of /MODIFY-SAT-SUPPORT-PARAMETERS.

In the same way, an alarm defined with /ADD or /MODIFY-SAT-ALARM-CONDITION is only triggered for a particular event if the event does not relate to a product for which alerting has been deactivated.

Currently, the activation and deactivation of SAT logging is only supported for POSIX.

MODIFY-**SAT-SUP**PORT-**PAR**AMETERS
POSIX-EVENTS = ***UNCHA**NGED / ***DISABLE**D / ***ENABLE**D

POSIX-EVENTS =
Specifies whether SAT logging and SAT alarms are to be activated or deactivated for POSIX (Portable Open System Interface for UNIX).

The events in question are the SAT object events POSIX-FILE-and-Directory, POSIX-PROCESS, POSIX-CHILD-Process and POSIX-SYSTEM-Resources.

POSIX-EVENTS = *UNCHANGED
The setting for the events of the specified product remains unchanged.

POSIX-EVENTS = *DISABLED
SAT logging and alarms are deactivated for the specified product. This means that no SAT logging is performed and no SAT alarms are triggered for events relating to this product.

Deactivation does not modify the preselection and alarm definitions. However, they are no longer effective for the events in question.

POSIX-EVENTS = *ENABLED
SAT logging and alarms are activated for the specified product. This means that SAT logging is performed for the events of these products in accordance with preselection and filter settings and that the alarms can be triggered in accordance with definitions.

The specification POSIX-EVENTS=*ENABLED only **enables** logging or alarm triggering for the corresponding events. To perform actual logging or issue alarms for these events, you must use the /MODIFY-SAT-PRESELECTION or /ADD-SAT-ALARM-CONDITIONS commands.

Activation does not modify the preselection settings or the alarm definitions. However, they are only effective for the events in question.

Notes

- By default, SAT support for POSIX is deactivated and must be explicitly activated in order to log POSIX events.
- Any modifications which the security administrator may make to the preselection default settings for POSIX events are independent of the SAT support setting and may be performed and saved at any time.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without errors
	32	SAT0000	Unrecoverable error SATCP possibly is in an incosistent state
	32	SAT5000	Nonrecoverable error
	64	SAT1000	User does not have privilege for this command
	64	SAT1050	Command not permitted if logging function activated
	130	SAT1010	Another command is currently being executed
	130	SAT1080	Change-over in preparation

Example

The security administrator activates SAT support for POSIX events:

```
/modify-sat-support-parameters posix-events=*enabled
```

2.5.10 REMOVE-SAT-ALARM-CONDITIONS Remove alarm definitions

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The /REMOVE-SAT-ALARM-CONDITIONS command can be used to delete existing alarm definitions.

REMOVE**-SAT-ALARM-COND**ITIONS
NAME = ***ALL** / list-poss(32): <name 1..8>

NAME = list-poss(32): <name 1..8> / *ALL
Name of the alarm to be deleted, as defined by means of /ADD-SAT-ALARM-CONDITIONS.

NAME = *ALL
All alarm definitions are to be deleted.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	64	SAT1021	Alarm already exists in alarm list
	64	SAT1028	Alarm unknown
	64	SAT1050	Command permitted only if logging function is activated
	64	SAT1070	Alarms table is empty
	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared

Note

See the general notes on SAT commands on "Functional overview".

2.5.11 REMOVE-SAT-FILTER-CONDITIONS Remove filter definitions

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The /REMOVE-SAT-FILTER-CONDITIONS command can be used to delete existing filter definitions.

REMOVE**-SAT-FILTER-COND**ITIONS
NAME = ***ALL** / list-poss(32): <name 1..8>

NAME = list-poss(32): <name 1..8> / *ALL
Name of the filter to be deleted, as defined by means of /ADD-SAT-FILTER-CONDITIONS.

NAME = *ALL
All filter definitions are to be deleted.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	64	SAT1031	Filter already exists in filter list
	64	SAT1032	Filter unknown
	64	SAT1050	Command permitted only if logging function is activated
	64	SAT1072	Filters table is empty
	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared

Note

See the general notes on SAT commands on "Functional overview".

2.5.12 RESUME-SAT-LOGGING Resume SAT logging

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The /RESUME-SAT-LOGGING command is used by the security administrator (user ID with the privilege SECURITY-ADMINISTRATION) to reactivate the logging of events and the SAT alarm function, both of which were previously suspended by means of the /HOLD-SAT-LOGGING command. A new SATLOG file is automatically opened at the same time.

RESUME**-SAT-LOG**GING

This command has no operands.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	128	SAT1010	Another command is currently being processed
	128	SAT1011	Processing of a /HOLD-SAT-LOGGING command not yet completed
	128	SAT1080	Exchange being prepared
	128	SAT2000	Logging function already activated
2	128	SAT2030	DMS error when opening file

Notes

1. The command is rejected if SATCP is not in the HOLD state.
2. On execution of the /RESUME-SAT-LOGGING command the same logging setting applies as before the suspension of SATCP.
3. Following successful execution of the command, SATCP is ready for auditing and will write its data to the new SATLOG file that has been opened.
4. The new SATLOG file is cataloged on the user-specified pubset of user ID SYSAUDIT. In the event of a DMS error during creation of the new SATLOG file (e.g. if the user-specified pubset is not available) the SATLOG file is created on the home pubset.
5. Due to the serialization of the SAT commands and the security gaps which could result, the /RESUME-SAT-LOGGING command should not be used to change the SATLOG file. The /CHANGE-SAT-FILE command is available for this purpose.

6. If the /RESUME-SAT-LOGGING command is entered immediately after the /HOLD-SAT-LOGGING command, an error may be indicated.
   Processing for the /HOLD-SAT-LOGGING command requires a certain amount of time in order to place the SAT environment asynchronously in HOLD status. Only when this state has been achieved is a resumption possible.
7. See also the general notes on SAT commands on "Functional overview".

2.5.13 SAVE-SAT-PARAMETERS Save SATCP settings

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION, SAT-FILE-MANAGEMENT

/SAVE-SAT-PARAMETERS permits the security administrator or the SAT file manager to save the EVENT-PRESELECTION settings, alarm and filter definitions, SAT support parameter settings and SATLOG file attributes for the next session. Missing or invalid values are set by SAT to the default values.

The SAT parameter file is always created on the home pubset and has the name: $SYSAUDIT.SYSPAR.SAT.

The default values are:

EVENT-PRESELECTION:	• Event: default setting of security-relevant events (see "Table of object-related events") • User IDs: all events are logged for newly created user IDs. • Logic rule: INDEPENDENT rule • Exit activation: system exit 110 not active • Recording scope: *EXTENDED fields are not recorded
FILTER-CONDITIONS:	no filters defined
ALARM-CONDITIONS:	no alarms defined
SAT-FILE-ATTRIBUTES:	BUFFER-LENGTH=*STD(SIZE=2), SPACE=*RELATIVE(PRIMARY-ALLOCATION=120, SECONDARY-ALLOCATION=120)
SAT-SUPPORT	POSIX-EVENTS=*DISABLED

SAVE-SAT-PARAMETERS
EVENT-PRESELECTION = ***UNCHA**NGED / ***STD** / ***CURR**ENT
,**ALARM-COND**ITIONS = ***UNCHA**NGED / ***STD** / ***CURR**ENT
,**SAT-FILE-ATTR**IBUTES = ***UNCHA**NGED / ***STD** / ***CURR**ENT
,**FILTER-COND**ITIONS = ***UNCHA**NGED / ***STD** / ***CURR**ENT
,**SAT-SUP**PORT = ***UNCHA**NGED / ***STD** / ***CURR**ENT

EVENT-PRESELECTION =
This operand may be specified only by the security administrator (privilege SECURITY-ADMINISTRATION).

EVENT-PRESELECTION = *UNCHANGED
EVENT-PRESELECTION in the SAT parameter file is not to be changed.

EVENT-PRESELECTION = *STD
The default values for EVENT-PRESELECTION are to be stored in the SAT parameter file. The default values are listed in the table “Object-related events, event names and auditattributes”.

EVENT-PRESELECTION = *CURRENT
The currently valid values are to be stored in the SAT parameter file. These values can be displayed with /SHOW-SAT-STATUS.

ALARM-CONDITIONS =
This operand may be specified only by the security administrator (privilege SECURITY-ADMINISTRATION).

ALARM-CONDITIONS = *UNCHANGED
The ALARM-CONDITIONS in the SAT parameter file are not to be changed.

ALARM-CONDITIONS = *STD
The default values for ALARM-CONDITIONS are to be stored in the SAT parameter file. This means that **no** alarm definitions are stored.

ALARM-CONDITIONS = *CURRENT
The currently valid values are to be stored in the SAT parameter file. These values can be displayed with /SHOW-SAT-ALARM-CONDITIONS.

SAT-FILE-ATTRIBUTES =
This operand is available to the SAT file manager only (SAT-FILE-MANAGEMENT privilege).

SAT-FILE-ATTRIBUTES = *UNCHANGED
The SAT-FILE-ATTRIBUTES in the SAT parameter file are not to be changed.

SAT-FILE-ATTRIBUTES = *STD
The default values for SAT-FILE-ATTRIBUTES are to be stored in the SAT parameter file (see the beginning of this section).

SAT-FILE-ATTRIBUTES = *CURRENT
The currently valid values are to be stored in the SAT parameter file.

FILTER-CONDITIONS =
This operand may be specified only by the security administrator (privilege SECURITY-ADMINISTRATION).

FILTER-CONDITIONS = *UNCHANGED
The FILTER-CONDITIONS in the SAT parameter file are not to be changed.

FILTER-CONDITIONS = *STD
The default values for FILTER-CONDITIONS are to be stored in the SAT parameter file. This means that **no** filter definitions are stored.

FILTER-CONDITIONS = *CURRENT
The currently valid values are to be stored in the SAT parameter file. These values can be displayed with SHOW-SAT-FILTER-CONDITIONS.

SAT-SUPPORT =
This operand is only available to security administrators (privilege SECURITY-ADMINISTRATION).

SAT-SUPPORT = *UNCHANGED
The SAT parameter file is not modified for SAT-SUPPORT.

SAT-SUPPORT = *STD
The default values are entered for SAT support in the SAT parameter file (see the start of this command description at).

SAT-SUPPORT = *CURRENT
The currently valid values are entered in the SAT parameter file and can be displayed using /SHOW-SAT-SUPPORT-PARAMETERS.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed SAT parameter file open Warning: error in checking
	32	CMD0221	Unrecoverable error DMS error SAT parameter file invalid Abnormal termination of SATCP Error during initialization of SATCP Error during initialization of SAVE/RESTORE
	64	SAT1000	User not privileged for command
	64	SAT1050	Command permitted only if logging function is activated
	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared

Notes

1. If *STD or *CURRENT is specified for one or more parameters, the values that were previously stored will be overwritten.
2. See also the general notes on the SAT parameter file on "SAT parameter file".
3. See also the general notes on SAT commands on "Functional overview".

2.5.14 SHOW-SAT-ALARM-CONDITIONS Display SAT alarm definitions

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

/SHOW-SAT-ALARM-CONDITIONS displays information about alarm definitions.

```
SHOW-SAT-ALARM-CONDITIONS

  NAME = *ALL / list-poss(32): <name 1..8>

  ,INFORMATION = *ALL / *NAME

  ,VALUE = *CURRENT / *STD / *NEXT-SESSION

  ,OUTPUT = *SYSOUT / *SYSLST(...)

      *SYSLST(...)

          |    LINES-PER-PAGE = 64 / <integer 20..255>
```

NAME = *ALL / list-poss(32): <name 1..8>
This defines the extent of the information to be displayed.

INFORMATION =
This specifies which information is to be output in relation to an alarm definition.

INFORMATION = *ALL
All information (name, definitions and response) is to be displayed.

INFORMATION = *NAME
Only the name of the definition is to be displayed.

VALUE =
This specifies which alarm definitions are to be output. The scope of the output will differ, depending on whether or not the current alarm definitions have already been saved in the SAT parameter file.

VALUE = *CURRENT
The current alarm definitions are to be displayed. If changes have been made to the alarm definitions since SATCP was started, but these have not been saved in the SAT parameter file, the definitions for the next session will differ from those for the current session because SATCP uses the definitions from the SAT parameter file when it is next started.

VALUE = *STD
The standard value for alarm definitions is to be output. By default, no alarm definitions exist at the present time.

VALUE = *NEXT-SESSION
This function shows the contents of the SAT parameter file. If changes have been made to the alarm definitions since SATCP was started, but these have not been saved in the SAT parameter file, the definitions for the next session will differ from those for the current session because SATCP uses the definitions from the SAT parameter file when it is next started.

OUTPUT = *SYSOUT
The requested information is to be sent to SYSOUT.

OUTPUT = *SYSLST(...)
The requested information is to be sent to SYSLST.

LINES-PER-PAGE = 64 / <integer 20..255>
This defines the number of lines on the output page.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	64	SAT1021	Alarm already exists in alarm list
	64	SAT1028	Alarm unknown
	64	SAT1070	No alarm is currently defined
	64	SAT1074	By default no alarm is defined
	64	SAT1075	No alarm is defined in the SAT parameter file
	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared
	128	SAT4010	SAT parameter file not currently available

Notes

- The command does not write to S variables since it can only be executed by the security administrator who, however, does not possess the STD-PROCESSING privilege which is required for the processing of S variables.

- The filpos, curlim2 and maxlim2 fields are always output in multiples of 512 bytes together with the unit “(512 B)“.

  Example

  The following alarm conditions are defined:

```
/add-sat-alarm-conditions alarm1, -
/        select=*par ( -
/          field-name=( -
/           *filpos(value=( -
/              512(unit=*kb),10240(unit=*bytes),6(unit=*mb))), -
/           *curlim(value=(513,10240,7)), -
/           *curlim2(value=( -
/              1024(unit=*kb),1536(unit=*bytes),2(unit=*mb)))))
```

  These alarm conditions are output as follows:

```
/show-sat-alarm-conditions alarm1
```

```
ALARM   NAME =  ALARM1      TIME-LIMIT = UNDEFINED       REPEAT =   3
        TRIGGER-ACTION = OPERATOR-MESSAGE (WAIT-RESPONSE = YES)
EVENTS   :   *ALL
USERS    :   *ALL
FIELD    :   CURLIM
             ONLY VALUES    : 7 / 513 / 10240
FIELD    :   FILPOS
             ONLY VALUES    : 20 (512B) / 1024 (512B) / 12288 (512B)
FIELD    :   CURLIM2
             ONLY VALUES    : 3 (512B) / 2048 (512B) / 4096 (512B)
```

- See the general notes on SAT commands in "Functional overview".

Example

The security administrator wants to output a list showing alarm definitions that have already been entered, for use in the next session.

```
/show-sat-alarm-conditions information=*name,value=*next-session
```

2.5.15 SHOW-SAT-FILTER-CONDITIONS Display SAT filter definitions

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

/SHOW-SAT-FILTER-CONDITIONS displays information about filter definitions.

```
SHOW-SAT-FILTER-CONDITIONS

  NAME = *ALL / list-poss(32): <name 1..8>

  ,INFORMATION = *ALL / *NAME

  ,VALUE = *CURRENT / *STD / *NEXT-SESSION

  ,OUTPUT = *SYSOUT / *SYSLST(...)

      *SYSLST(...)

          |    LINES-PER-PAGE = 64 / <integer 20..255>
```

NAME = *ALL / list-poss(32): <name 1..8>
This defines the extent of the information to be displayed.

INFORMATION =
This specifies which information is to be output in relation to an filter definition.

INFORMATION = *ALL
All information (name, definitions and response) is to be displayed.

INFORMATION = *NAME
Only the name of the definition is to be displayed.

VALUE =
This specifies which filter definitions are to be output. The scope of the output will differ, depending on whether or not the current filter definitions have already been saved in the SAT parameter file.

VALUE = *CURRENT
The current filter definitions are to be displayed. If changes have been made to the filter definitions since SATCP was started, but these have not been saved in the SAT parameter file, the definitions for the next session will differ from those for the current session because SATCP uses the definitions from the SAT parameter file when it is next started.

VALUE = *STD
The standard value for filter definitions is to be output. By default, no filter definitions exist at the present time.

VALUE = *NEXT-SESSION
This function shows the contents of the SAT parameter file. If changes have been made to the filter definitions since SATCP was started, but these have not been saved in the SAT parameter file, the definitions for the next session will differ from those for the current session because SATCP reads the definitions from the SAT parameter file when it is next started.

OUTPUT = *SYSOUT
The requested information is to be sent to SYSOUT.

OUTPUT = *SYSLST(...)
The requested information is to be sent to SYSLST.

LINES-PER-PAGE = 64 / <integer 20..255>
This defines the number of lines on the output page.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed
	32	SAT0000	Unrecoverable error
	64	SAT1000	User not privileged for command
	64	SAT1031	Filter already exists in filter list
	64	SAT1032	Filter unknown
	64	SAT1072	No filter is currently defined
	64	SAT1076	By default no filter is defined
	64	SAT1077	No filter is defined in the SAT parameter file
	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared
	128	SAT4010	SAT parameter file not currently available

Notes

- This command does not write any S variables. The command may only be used under a user ID with the privilege SECURITY-ADMINISTRATION. Such a user ID does not, however, have the privilege STD-PROCESSING required for SDF-P.

- The filpos, curlim2 and maxlim2 fields are always output in multiples of 512 bytes together with the unit “(512 B)“.

  Example

  The following filter conditions are defined:

```
/add-sat-filter-conditions filter1, -
/        select=*par ( -
/          field-name=( -
/           *filpos(value=( -
/              512(unit=*kb),10240(unit=*bytes),6(unit=*mb))), -
/           *curlim(value=(513,10240,7)), -
/           *curlim2(value=( -
/              1024(unit=*kb),1536(unit=*bytes),2(unit=*mb)))))
```

  These filter conditions are output as follows:

```
/show-sat-filter-conditions filter1
```

```
FILTER   NAME =  FILTER1
        TRIGGER-ACTION = LOGGING (RECORDING = YES)
EVENTS   :   *ALL
USERS    :   *ALL
FIELD    :   CURLIM
             ONLY VALUES    : 7 / 513 / 10240
FIELD    :   FILPOS
             ONLY VALUES    : 20 (512B) / 1024 (512B) / 12288 (512B)
FIELD    :   CURLIM2
             ONLY VALUES    : 3 (512B) / 2048 (512B) / 4096 (512B)
```

- See the general notes on SAT commands in "Functional overview".

Example

The security administrator wishes to output a list containing the filter definitions already entered which he/she intends to use in the next session.

```
/show-sat-filter-conditions information=*name,value=*next-session
```

2.5.16 SHOW-SAT-STATUS Output SAT status

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION, SAT-FILE-MANAGEMENT

The /SHOW-SAT-STATUS command can be used by the security administrator and the SAT file manager to request the output of information on SAT auditing.

```
SHOW-SAT-STATUS

INFORMATION = *SUMMARY / list-poss(5): *LOGGING-STATUS / *COLLECTION-FILE /
                        *PRESELECTION-RULE / *EVENT-AUDITING(...) / *USER-AUDITING(...)
    *EVENT-AUDITING(...)
        | EVENT-NAME = *ALL(...) / list-poss(50): <name 3..3>
        |     *ALL(...)
        |         | AUDIT-SWITCH = *IGNORE / *ON(...) / *OFF
        |         |     *ON(...)
        |         |         | RESULT = *ALL / *SUCCESS / *FAILURE
    *USER-AUDITING(...)
        | USER-IDENTIFICATION = *ALL-SWITCHABLE(...) / *ALL(...) / list-poss(50): <name 1..8>
        |     *ALL-SWITCHABLE(...)
        |         | AUDIT-SWITCH = *IGNORE / *ON / *OFF
        |     *ALL(...)
        |         | AUDIT-SWITCH = *IGNORE / *ON / *OFF
,OUTPUT = *SYSOUT / *SYSLST(...)
    *SYSLST(...)
        | LINES-PER-PAGE = 64 / <integer 20..255>
,VALUE = *CURRENT / *STD / *NEXT-SESSION
```

INFORMATION =
This defines the type of information to be output.

INFORMATION = *SUMMARY
The information described under LOGGING-STATUS, COLLECTION-FILE and PRESELECTION-RULE is to be output. For reasons of compatibility the value *STD is still supported when used instead of *SUMMARY.

INFORMATION = *LOGGING-STATUS
The current SAT status (RECORD, HOLD, NO RESOURCE, SHUTDOWN) is to be output.

INFORMATION = *COLLECTION-FILE
The attributes of the SATLOG file (Name, SUPPORT, BUFFER-LENGTH, PRIMARY-ALLOCATION and SECONDARY-ALLOCATION) are to be output.

INFORMATION = *PRESELECTION-RULE
Displays the following information:

- the current selection rule: *INDEPENDENT or *FILES-BY-EVENTS
- the recording scope: *STD or *EXTENDED
- the EXIT activation: *YES or *NO.
- the default value for the audit setting for newly created user IDs: *ON or *OFF.

INFORMATION = *EVENT-AUDITING(...)
Information about events is to be output.

EVENT-NAME = *ALL(...)
The events to which a certain auditing setting applies are to be output.

AUDIT-SWITCH = *IGNORE
A list of all events is to be output, irrespective of whether or not they have been selected for auditing.

AUDIT-SWITCH = *ON(...)
A list of those events is to be output which have been selected for auditing. This includes permanent selection (audit state *ON) as well as temporary selection (audit state ON).

RESULT = *ALL / *SUCCESS / *FAILURE
A list of those events is to be output which have been selected for auditing and whose audit attributes match the specified RESULT value.

AUDIT-SWITCH = *OFF
A list of all events not selected for auditing is to be output.

EVENT-NAME = <name 3..3>
The auditing settings for the specified events are to be output. The entry consists of the 3-character name of the event type, e.g. FCD, FRD,... (see “Table of object-related events”).

INFORMATION = *USER-AUDITING(...)
The users selected for auditing are to be output.

USER-IDENTIFICATION = *ALL-SWITCHABLE(...)
All user IDs are to be output for which auditing may be activated/deactivated (“switchable” user IDs) and which have a specific audit attribute (the user ID of the security administrator, SYSAUDIT and user IDs possessing the SAT-FILE-
MANAGEMENT privilege are not switchable).

AUDIT-SWITCH = *IGNORE / *ON / *OFF
The audit attribute of the “switchable” user IDs is to be either applicable/not applicable or to be ignored (default value).

USER-IDENTIFICATION = *ALL(...)
All user IDs with a specific audit attribute are to be output.

AUDIT-SWITCH = *IGNORE / *ON / *OFF
The audit attribute of the user IDs is to be either applicable/not applicable or to be ignored (default value).

USER-IDENTIFICATION = <name 1..8>
The auditing settings for the specified user ID are to be output.

OUTPUT = *SYSOUT
The requested information is to be output to SYSOUT.

OUTPUT = *SYSLST(...)
The requested information is to be output to SYSLST.

LINES-PER-PAGE = 64 / <integer 20..255>
This defines the number of lines on the output page.

VALUE = *CURRENT / *STD / *NEXT-SESSION
This specifies which information is to be output:
the currently valid values, the default values or the values which will be valid in the next session. The list of user IDs is not output unless INFORMATION=*USER-AUDITING and VALUE=*CURRENT have been specified.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command successfully executed
	32	SAT0000	Unrecoverable error
	32	CMD2009	System error during output of S variables
	64	SAT1000	User not privileged for command
	64	SAT1020	Event already exists in event list
	64	SAT1030	User already exists in user list
	64	SAT1040	Information already exists in information list
	64	SAT1060	No information available
	64	OPS0002	Output of S variables has been aborted
	128	SAT1010	Another command is currently being processed
	128	SAT1080	Exchange being prepared
	128	SAT4010	SAT parameter file not currently available
	130	OPS0001	It was not possible to output the S variables

Notes

1. An error message is returned if one or more of the specified user IDs does not exist in the user catalog. The command is executed for those user IDs present in the user catalog. The same rule applies to unknown event types.
2. The command is rejected if the list of events contains the same event more than once. The same rule applies to the list of user IDs.
3. If the audit setting for an event or a user is preceded by an asterisk (*) this setting cannot be altered.

   Example: **USER-AUDITING**

   SYSAUDIT *ON SYSPRIV *ON TSOS OFF

4. See also the general notes on SAT commands in "Functional overview".

Examples

1. The security administrator wishes to have information about the current status of SAT, about the assigned SATLOG file and the valid linkage rule:

   /show-sat-status

   This command provides the following output:

```
SAT SUBSYSTEM VERSION 05.6A10                              VALUE = CURRENT
 LOGGING-STATUS        : RECORD
 COLLECTION-FILE(SATLOG) :
 FILENAME : :A:$SYSAUDIT.SYS.SATLOG.2021-10-06.003.01
 STATUS   : OPENED
 BLOCK    : (STD,2)
 SPACE    : (1002,1002)
 REPEAT   : NO
 WAITING FOR PUBSET    : *STD
 PRESELECTION-RULE     : INDEPENDENT
 BY-EXIT               : NO
 LOGGING-QUANTITY      : STD
 USER-AUDITING DEFAULT : ON
```

   Waiting for pubset outputs:
   • *STD : Standard processing. Home pubset will be preferred for SATLOG.
   • *NO : SATLOG is already located on selected pubset.
   • <pubset_name> : SATLOG will move if named pubset is imported.

2. The security administrator wishes to restrict the output to those events explicitly selected for auditing (RESULT = ALL). This is done by entering the following command:

   /show-sat-status information= -
   / *event-auditing(event-name=*all(audit-switch=*on(result=*all)))

   or its abbreviated form:

   /show-sat-stat inf=*event-audit(event-name=*all(audit-switch=*on))

3. The security administrator wishes to

   - output the selection parameters for all events
   - output information on the selection parameters for the user IDs BILL, HUGO and JAMES.

   This is done by entering the following command:

```
/show-sat-status information=( -
/      *event-auditing(event-name=*all), -
/      *user-auditing(user-identification=(bill,hugo,james))
```

or its abbreviated form:

```
/show-sat-stat (event-audit,user-audit((bill,hugo,james)))
```

Output in S variables

The INFORMATION operand of this command specifies which S variables are assigned values. The possible entries for INFORMATION are as follows:

Notation in command	Abbreviated notation in table
INFORMATION = SUMMARY	1
INFORMATION = LOGGING-STATUS	2
INFORMATION = COLLECTION-FILE	3
INFORMATION = PRESELECTION-RULE	4
INFORMATION = EVENT-AUTDITING	5
INFORMATION = USER-AUDITING	6

Additional conditions which interact with the specifications of INFORMATION:

Additional conditions	Abbreviated notation in table
Value assignment, only if LOG-F.REPEAT=*TRUE	b

The table below is arranged according to the names of the S variables. Column T (type) indicates the data type of the contents: S (string), I (integer), B (boolean).

Output information	Name of the S variable	T	Contents	Condition
Audit attribute of the event in the SATLOG file which determines whether an event is selected for auditing	var(*LIST).EVENT-AUDIT(*LIST). AUDIT-SWITCH	S	*OFF *ON-ALL *ON-FAIL *ON-SUCC	5
Abbreviated name of the event type for which the auditing setting is displayed	var(*LIST).EVENT-AUDIT(*LIST). EVENT-NAME	S	<name 3..3>	5
Event has been selected for auditing	var(*LIST).EVENT-AUDIT(*LIST). SWITCHABLE	B	FALSE TRUE	5
Buffer size of the SATLOG file	var(*LIST).LOG-F.BUF-LEN	I	<integer 1..16> <buffer-length>	1,3

Name of the SATLOG file	var(*LIST).LOG-F.NAME	S	<filename>	1,3
SATLOG file is open	var(*LIST).LOG-F.OPEN	B	FALSE TRUE	1,3
Time period (in days) after which automatic switching of the SATLOG file takes place	var(*LIST).LOG-F.PERIOD-DAYS	I	<integer 0..10>	b
Time period (in hours) after which automatic switching of the SATLOG file takes place	var(*LIST).LOG-F.PERIOD-HOURS	I	<integer 0..23>	b
Primary storage space allocation for the SATLOG file	var(*LIST).LOG-F.PRIMARY-ALLOC	I	<integer>	1,3
The logging file is changed automatically	var(*LIST).LOG-F.REPEAT	B	FALSE TRUE	1,3
Secondary storage space allocation for the SATLOG file	var(*LIST).LOG-F.SECONDARY-ALLOC	I	<integer 0.. 32767>	1,3
Type of disk storage on which the SATLOG file is stored	var(*LIST).LOG-F.SUP-TYPE	S	*PUBLIC	1,3
Desire to move SATLOG file to another pubset	Var(*LIST).LOG-F.WAITING-FOR-PUBSET	S	*STD *NO <pubset_name>	1,3
Current SAT status	var(*LIST).LOG-STA	S	*HOLD *NO-RESOURCE *REC	1,2
Exit routine 110 can be invoked	var(*LIST).PRESEL-RULE.EXIT	S	*NO *YES	1,4
Logging quantity	var(*LIST).PRESEL-RULE.QUANTITY	S	*STD *EXTENDED	1,4
Type of logic rule for logging the event	var(*LIST).PRESEL-RULE.RULE	S	*FILES-BY-EVENTS *INDEPENDENT	1,4
Default value for the audit setting for newly created user IDs	var(*LIST).PRESEL-RULE.USER-AUDIT-DEF	S	*OFF *ON	1,4
Audit attribute of the subject which determines whether the events relating to this user ID are to be audited	var(*LIST).USER-AUDIT(*LIST).AUDIT-SWITCH	S	*OFF *ON	6

Indication of whether the user ID is switchable	var(*LIST).USER-AUDIT(*LIST). SWITCHABLE	B	FALSE TRUE	6
User ID for which auditing is activated	var(*LIST).USER-AUDIT(*LIST). USER-ID	S	<name 1..8>	6

2.5.17 SHOW-SAT-SUPPORT-PARAMETERS Display parameters for product-specific logging and alarms

Domain:	SECURITY-ADMINISTRATION
Privileges:	SECURITY-ADMINISTRATION

The security administrator, the SAT file manager and the SAT file analyst can use the command /SHOW-SAT-SUPPORT-PARAMETERS to find out which product SAT logging and the triggering of SAT alarms have been activated or deactivated.

```
SHOW-SAT-SUPPORT-PARAMETERS

  VALUE = *CURRENT / *STD / *NEXT-SESSION

 ,OUTPUT = *SYSOUT / *SYSLST(...)

     *SYSLST(...)

         |    LINES-PER-PAGE = 64 / <integer 20..255>
```

VALUE =
Specifies what information is output.

VALUE = *CURRENT
The currently valid values are output.

VALUE = *STD
The default values are output.

VALUE = *NEXT-SESSION
The values that will apply in the next session are output.

OUTPUT = *SYSOUT
The requested information is output to SYSOUT.

OUTPUT = *SYSLST(...)
The requested information is output to SYSLST.

LINES-PER-PAGE = 64 / <integer 20..255>
Specifies the number of lines on an output page.

Command return codes

(SC2)	SC1	Maincode	Meaning
	0	CMD0001	Command executed without errors
	32	CMD2009	System error during S variable output
	32	SAT0000	Unrecoverable error SATCP possibly is in an incosistent state

	32	SAT5000	Nonrecoverable error
	64	OPS0002	Output of S variables was interrupted
	64	SAT1000	User does not have the privilege for this command
	64	SAT5001	Insufficient memory allocation for SYSLST file
	130	CMD2009	OPS not available
	130	OPS0001	Not possible to perform output in S variables
	130	SAT1010	Another command is currently being executed
	130	SAT1080	Change in preparation
	130	SAT4010	SAT parameter file is not currently available

Output in S variables

Output information	Name of the S variable	T	Contents	Condition
SAT support for POSIX events	var(*LIST).POSIX-EVENTS	S	*ENABLED *DISABLED	

Example

The user modifies SAT support for POSIX events and then wants information concerning the implemented changes.

```
/modify-sat-support-parameters posix-events=*enabled
/show-sat-support-parameters

SAT support for POSIX events : ENABLED
```

2.6 SATUT - evaluating SATLOG files

Editing the SATLOG files is the task of SAT file management or SAT file evaluation. The utility routine used for evaluation is SATUT, under the SYSAUDIT user ID.
It is executable independently of the SAT subsystem SATCP under any user ID that has the SAT-FILE-MANAGEMENT or SAT-FILE-EVALUATION privilege.
In addition to SATLOG files, SATUT can also include CONSLOG files (see "Tables of auditable information on object-related events (1)") in the evaluation.

The SAT evaluation routine SATUT offers the following functions:

- It uses the input files to create edited files (replacement files) containing the securityrelevant data selected by SAT file management or SAT file evaluation. In this case the function of SAT is to reduce the volume of data and store security-relevant records, i.e. the input files can be replaced by the edited files.
- It selects specific audit records from the input files on the basis of certain selection conditions. The selected records are then output to printer (SYSLST) or an XML file, presented as statistics or written to a file (analysis file). In this case, the function of SATUT is to analyze selected event groups, i.e. the input files are not replaced by the edited files.

2.6.1 Working with SATUT

The following basic functions, in the form of statements, are available for creating the edited logging files:

1. A statement for specifying the input files for the SATUT session (//SELECT-INPUT-FILES).
2. Statements for specifying selection conditions for editing (//ADD-, //REMOVE-, //SHOW-SELECTION-CONDITIONS).
3. A statement for implementing selection depending on the selection conditions (//START-SELECTION).
4. Statements by means of which the selected records are output to a printer (SYSLST), an XML file or in the form of statistics (//SHOW-SELECTED-RECORDS, //SHOW-STATISTICS).
5. A statement by means of which the selected records can be written to reduced SAT logging files (//SAVE-SELECTED-RECORDS).

The order of the functions corresponds to the way that an evaluation is organized with SATUT. Within this, functions 2. to 5. can be executed more than once.

2.6.2 Input files for SATUT

The following files are accepted as input files for the SAT evaluation routine SATUT:

- SATLOG files ($SYSAUDIT.SYS.SATLOG.yyyy-mm-dd.sss.nn)
- reduced SAT logging files with standard names (replacement files) ($SYSAUDIT.SYS.SATUT.yyyy-mm-dd.sss.nnn)
- CONSLOG files (with standard names)

Simultaneous input of replacement files and one or more input files from which they were derived is not permissible, as overlapping may occur.

As an alternative to the above list it is also possible to specify reduced SAT logging files without standard names (analysis files).
When analysis files are used for input it is possible that the same records occur more than once, falsifying the analysis results (particularly in the case of statistics).

2.6.3 Work files in the SATUT session

The records that are to be selected are initially buffered in a SATUT work file with the //START-SELECTION command for each selection condition. Up to ten such work files (0..9) are available in one SATUT session.

The contents of the work files can then be used as appropriate:

- output (see next section) or
- used as input for another START-SELECTION command

2.6.4 Output from SATUT

The selected records can be output in the following ways:

- in readable form to SYSLST or an XML file (//SHOW-SELECTED-RECORDS)
- in statistical form to SYSOUT or SYSLST (//SHOW-STATISTICS)
- in their original form to reduced SAT logging files with standard names (replacement files) or without standard names (analysis files) (//SAVE-SELECTED-RECORDS)

SATUT creates two types of reduced SAT logging files for archiving and analysis of security-relevant data: **replacement files** and **analysis files**.

Replacement files and analysis files contain the same kind of information. They are the result of one or more selection processes in a SATUT session.They differ in terms of their intended use and their nomenclature.

In addition to the user data both types of file contain additional information which can be output by means of the //SHOW-REDUCTION-FILES-ORIGIN statement:

- date of file creation
- the selection condition
- the input files from which the records were selected

Replacement files

Replacement files contain the security-relevant information from the input files selected by SAT file management or SAT file evaluation for **archiving**.
Replacement files are used for storing security-relevant audit records (SATLOG and also in converted form CONSLOG) and, if appropriate, input again in another evaluation run.

These files normally replace the input files from which they were generated. The SAT file manager or SAT file evaluator can decide whether to delete the input files when a replacement file is created, if dealing here exclusively with SATLOG files.
When replacement files replace the input files they should always replace complete SATLOG files or replacement files.

A replacement file is stored under the SYSAUDIT user ID with the SAVE-SELECTED-RECORDS statement. In that case the SATUT session must also be running under SYSAUDIT.

Replacement files have a standard name:

$SYSAUDIT.SYS.SATUT.yyyy-mm-dd.sss.nnn where:

yyyy-mm-dd	creation date of the first (i.e. “oldest”) of the input files used to produce the replacement file. The input files may be: SATLOG files, replacement files, CONSLOG files
sss	session number
nnn	sequence number of the file (001..999)

Analysis files

Analysis files contain the security-relevant information from the input files which have been selected for **analysis** by SAT file management or SAT file evaluation.
Analysis files are used for the decentralized analysis of security-relevant audit records (SATLOG and also in converted form CONSLOG).

Analysis files do not replace the input files from which they are generated.

The //SAVE-SELECTED RECORDS statement is used to store an analysis file under the user ID under which the SATUT session is running.
In contrast with a replacement file, therefore, an analysis file can be created under any other user ID with the SAT-FILE-MANAGEMENT or SAT-FILE-EVALUATION privilege.

Any name can be chosen for analysis files.

2.6.5 Starting SATUT

SATUT is started by means of the /START-SATUT command and terminated by means of the END statement. For reasons of compatibility the /START-EVALUATOR, /START-EVALU, /EVALUATOR and /SATUT commands are still supported for starting SATUT.

2.6.6 START-SATUT Initiate the evaluation of SATLOG files

Domain:	UTILITIES, SECURITY-ADMINISTRATION
Privileges:	SAT-FILE-MANAGEMENT, SAT-FILE-EVALUATION

START-SATUT
VERSION = ***STD** / <product-version>
,MONJV = ***NONE** / <filename 1..54 without-gen-vers>
,**CPU-LIMIT** = ***JOB-REST** / <integer 1..32767 seconds>

VERSION = *STD / <product-version>
The version of SATUT that is to be started.

VERSION = *STD
The version set by means of the /SELECT-PRODUCT-VERSION command is used as the default version.

VERSION = <product-version>
Explicit specification of the version.

MONJV = *NONE / <filename 1..54 without-gen-vers>
Specification of a monitor job variable for monitoring the SATUT session.

MONJV = *NONE
No monitor job variable is to be used.

MONJV = <filename 1..54 without-gen-vers>
The name of the job variable to be used.

CPU-LIMIT = *JOB-REST / <integer 1..32767 seconds>
Maximum CPU time, in seconds, which the program is allowed to use.

CPU-LIMIT = *JOB-REST
The remaining CPU time is to be used for the task.

CPU-LIMIT = <integer 1..32767 seconds>
Only the specified amount of CPU time is to be used.

Note

The monitoring job variable can assume the following values:

0000	No error
1010	Invalid statement or unexpected END statement. Events may be invalid or incomplete.
1020	User does not have the privilege to start SATUT
1030	Input files not present

2010	Invalid statement. Events may be invalid or incomplete.
2015	Unexpected end of file to SYSDTA, SATUT terminated
3020	Internal inconsistency, SATUT terminated with dump

2.6.7 SATUT statements

The following sections first provide a functional overview of all SATUT statements and then go on to describe the individual statements in alphabetical order. Each statement description starts with a general explanation of the function of the statement, followed by the statement format and a description of the various operands and their values. Where appropriate, the description of the operands is followed by an example of statement application.

The statement metasyntax is explained in the appendix.

2.6.8 Functional overview

Starting and terminating SATUT

START-SATUT (command)	Initiate the evaluation of SATLOG files
END	Terminate evaluation

Defining input files

SHOW-REDUCTION-FILES-ORIGIN	Show the origin of replacement files
SELECT-INPUT-FILES	Define the input files for SATUT

Defining selection conditions and selecting records

ADD-SELECTION-CONDITIONS	Define selection conditions
REMOVE-SELECTION-CONDITIONS	Delete selection conditions
SHOW-SELECTION-CONDITIONS	Display selection conditions
START-SELECTION	Perform selection with consideration of selection conditions

Saving and analyzing edited files

SAVE-SELECTED-RECORDS	Save the hit records in replacement/analysis files
SHOW-SELECTED-RECORDS	Output the hit records to SYSLST or an XML file
SHOW-STATISTICS	Output the statistics to SYSLST or SYSOUT

Selecting records according to selection conditions

(Still supported for reasons of compatibility only; the statements to be used instead are //ADD-SELECTION-CONDITIONS, //REMOVE-SELECTION-CONDITIONS, //SHOW-SELECTION-CONDITIONS and //START-SELECTION)

SELECT-RECORDS	Define editing condition Select records according to certain editing conditions

Sequence of statements

It is essential to pay attention to the order of input for the following statements, because one follows on from the other. Steps 2-4 can be specified more than once in any one evaluation run. The other SATUT statements can be specified as required in the evaluation run.

1. //SELECT-INPUT-FILES for defining the input data.
2. //ADD-SELECTION-CONDITIONS for defining the selection conditions.
3. //START-SELECTION for implementing data selection.

4. //SHOW-SELECTED-RECORD, //SHOW-STATISTICS, //SAVE-SELECTED-RECORDS for examining and saving the selected data.

2.6.9 ADD-SELECTION-CONDITIONS Define selection conditions

The //ADD-SELECTION-CONDITIONS statement is used to define and name a selection condition for audit records. Wildcard syntax is allowed in the selection condition.

ADD-SELECTION-**COND**ITIONS
NAME = <name 1..8> , **COND**ITION = ***NONE** / <text 1..1800 with-low>

NAME = <name 1..8>
The name of the selection condition that is defined in the CONDITION operand.

CONDITION = *NONE / <text 1..1800>
The selection conditions are defined here.

CONDITION = *NONE
Selection is unrestricted.

CONDITION = <text 1..1800 with-low>
The editing condition is composed of one or more logical expressions which are ANDed, ORed or linked by a NOT operation. In addition, the order in which the expressions are evaluated can be determined by means of parentheses “(...)”. The logical operations are applied to expressions which may be either “TRUE” or “FALSE” (see truth tables, below). Those records which fulfill the condition are selected.

The editing condition is specified as follows for <text 1..1800 with-low>:

```
[NOT] cond1 [{ OR/AND [NOT] cond2} ...]
```

The editing condition can be used for:

1. searching within a list
2. searching within a range
3. comparison with a specific value
4. verifying the existence of a specific field
5. searching with wildcard syntax

> In general, uppercase and lowercase are handled in the same way in the editing condition. A distinction is made only in the case of the values for the following field names: homedir, linknam, newpath, pathnam, shell and symbdev.

’cond’ may be:

1. searching within a list

   ```
   field-name IN-LIST/NOT-IN-LIST (value,...)
   ```

   field-name IN-LIST (value1,...valuen)
   A record is selected if the specified field exists and its contents match any of the specified values.

field-name NOT-IN-LIST (value1,...valuen)
A record is selected if the specified field does not contain any of the specified values or if the specified field does not exist.

2. searching within a range

   ```
   field-name IN-RANGE/NOT-IN-RANGE (value-range)
   ```

   field-name IN-RANGE (value:value)
   A record is selected if the specified field exists and its contents match any of the values within the specified range. Only the timestp field (format: yyyy-mm-dd/hh:mm:ss) and fields with the SDF data type integer are accepted.

   field-name NOT-IN-RANGE (value:value)
   A record is selected if the specified field does not contain any of the values within the specified range or if the specified field does not exist. Only the timestp field (format: yyyy-mm-dd/hh:mm:ss) and fields with the SDF data type integer are accepted.

3. comparison with a specific value

   ```
   field-name EQUAL/NOT-EQUAL value
   ```

   field-name EQUAL value
   A record is selected if the specified field exists and contains the specified value.

   field-name NOT-EQUAL value
   A record is selected if the specified field contains a value other than the specified value or if the field does not exist.

4. searching for a specific field name

   ```
   field-name PRESENT
   ```

   All records containing the specified field are to be selected.

5. searching with wildcard syntax

   ```
   field-name MATCH/NOT-MATCH pattern
   ```

   field-name MATCH pattern
   All records which match the specified search pattern are selected. Only fields with the SDF data type c-string, with the exception of plamrc, are accepted.

   field-name NOT-MATCH pattern
   All records which do not match the specified search pattern are selected. Only fields with the SDF data type c-string, with the exception of plamrc, are accepted.

Definitions

field-name

defines the types of auditable information, e.g. access, acckey,... (see table on "Table of auditable information (field names)"). All other specifications are interpreted as errors and rejected. Specification of the name of a *LNG field (see "Structure of the SATLOG files") for field-name is not permitted.

value

corresponds to the data types defined in SDF: <x-string>, <name>, <c-string>, <integer>, <keyword>. 'value' must be of the data type specified for the appropriate field name (see table on "Table of auditable information (field names)"). For instance, if 'field-name' is DMSRC, then 'value' must be of type x-string.

Special features with field-name = filpos / curlim2 / maxlim2:

A special data type <integer-with-unit> exists for the filpos, curlim2 and maxlim2 file names.This differs from data type <integer> in that a unit value can be entered in parentheses, i.e. <integer>(<unit>). Where <unit> can be BYTES, KB (=Kilobytes), MB (=Megabytes) or GB (=Gigabytes). BYTES is assumed if the entry is omitted.

- If BYTES is defined either explicitly or implicitly the numerical value must be a multiple of 512. Otherwise, the statement is rejected with an error message.
- A numerical value specified with a unit is always converted internally into multiples of 512 bytes. Only this value is relevant for the result of a selection condition, but not the form of the entry. For example, the entries `3145728(BYTES)`, `3072(KB)` and `3(MB)` are taken to be equal since they each represent the same value of 3145728 bytes.
- The maximum value of 2^{40}-512 (=1 099 511 627 264) bytes may not be exceeded, regardless of the UNIT entry. This results in the following maximum values, depending on the UNIT entry:

UNIT	Maximum numerical value	Corresponds in bytes to
BYTES	2^{31}-1 = 2 147 483 647	2^{31}-1 = 2 147 483 647
KB	2^{30}-1 = 1 073 741 823	2^{40}-2^{10} = 1 099 511 626 752
MB	2^{20}-1 = 1 048 575	2^{40}-2^{20} = 1 099 510 579 200
GB	2^{10}-1 = 1 023	2^{40}-2^{30} = 1 098 437 885 952

value-range

defines a value range in the following format: <value:value>.

pattern

identifies a c-string in which parts of the character string can be replaced by wildcards, in the same way as the SDF data type <c-string with-wild (n)>.
pattern may be up to 281 characters long.
The wildcard characters that may be used are as follows:

*	Replaces an arbitrary (even empty) character string
/	Replaces exactly one freely selectable character
\	Disables wildcards (* / < > : ,) in a character string (e.g. ab*c identifies the string “ab*c”)

<s_x:s_y>	Replaces a string that meets the following conditions: • at least as long as the shortest string (s_x or s_y) • not longer than the longest string (s_x or s_y) • between s_x and s_y in the alphabetic collating sequence; numbers are sorted after letters (A... Z 0...9) • s_x may also be an empty string, which is in the first position in the alphabetic collating sequence • s_y may also be an empty string, which stands at this position for the string with the highest possible coding (contains only the characters X‘FF‘) • s_x must be before s_y in the alphabetic collating sequence. If s_x isshorter than s_y, s_x is filled with X‘00‘ • if s_y is shorter than s_x, s_y is filled with X‘FF‘ • wildcards are not allowed either in s_x or in s_y
<s1,...>	Replaces all strings matching any of the character combinations specified by s. s may also be an empty string. Any such string may also be a range specification <s_x:s_y>

The wildcard “-” for negating information is not used here.
NOT-MATCH is provided for this purpose.

Notes

1. If the syntax analysis of the statement detects an error, the editing condition is output to SYSOUT. The error is marked by a question mark.
2. In guided dialog, this output to SYSOUT causes the SDF screen to be lost; it can be restored by means of the //RESTORE-SDF-INPUT statement.
3. Posix filenames und Kerberos names are logged by SAT without any restriction. The following SAT fields are case-sensitive in the definition of selection conditions: AUDITID, HOMEDIR, LINKNAM, NEWPATH, PATHNAM, PRINCCL, PRINCSV, SHELL, SYMBDEV. The fields that are not case-sensitive are internally converted into uppercase letters. With the exception of SYMBDEV, however, these field can be specified with a maximum length of 255 bytes only. Events with longer field contents may be specified by using wildcards. In the specification of a single name (without wildcard) the same special characters are allowed as for posix filenames or Kerberos names.

Examples

```
//add-selection-conditions name = filesel, -
//                    condition = filname in-list ('filex','filey') - —  (1)
//                            and access equal input - ————————————————  (2)
//                            and res    equal f - ————————————————————  (3)
//                            and dmsrc  equal x'0d35' - ——————————————  (4)
```

Under the condition named filesel, events are selected if

(1) the file 'FILEX' or 'FILEY' is affected **and**

(2) the open mode is INPUT **and**

(3) the operation result is 'failure' **and**

(4) the file is not shareable (DMS return code 0D35).

```
//add-selection-conditions name = groupsel,  -
//                    condition = (groupid equal c'g1' - ——————————————  (1)
//                              and not auditid present) - ——————————————  (2)
//                              or (groupid in-list (c'g2',c'g3') - ———  (3)
//                                  and user-id not-in-list ('u1','u2'))  (4)
```

Under the condition named groupsel, events are selected if

(1) they were generated by users with group ID G1 **and** the users did **not**

(2) use a chipcard to identify themselves to the system **or**

(3) they were generated by users with group ID G2 OR G3 **and**

(4) these users did not have the user ID U1 or U2.

```
//add-selection-conditions name = satsel, -
//   condition = evt equal 'FRD' -
//         and filname match '$sysaudit.sys.satlog.*' -
//         and timestp in-range (2017-05-01/00:00:00 : 2017-05-31/23:59:59) -
//         and userid not-in-list ('tsos','sysaudit')
```

Read accesses by nonprivileged user IDs to SATLOG files in May 2017 are selected under the condition with the name satsel.

Truth tables

The following truth tables apply to AND, OR and NOT:

cond1 AND cond2	**TRUE**	**FALSE**
TRUE	TRUE	FALSE
FALSE	FALSE	FALSE

cond1 OR cond2	**TRUE**	**FALSE**
TRUE	TRUE	TRUE
FALSE	TRUE	FALSE

cond1	**TRUE**	**FALSE**
NOT cond1	FALSE	TRUE

Identical operators are processed from left to right.

2.6.10 END Terminate evaluation

The //END statement terminates evaluation with SATUT.

```
END
```

This statement has no operands.

2.6.11 REMOVE-SELECTION-CONDITIONS Remove selection conditions

The //REMOVE-SELECTION-CONDITIONS statement is used to remove one or more selection conditions (see the //ADD-SELECTION-CONDITIONS command).

REMOVE-**SEL**ECTION-**COND**ITIONS
NAME = ***ALL** / list-poss(10): <name 1..8>

NAME = *ALL / list-poss(10): <name 1..8>
Specification of the selection conditions that are to be removed.

NAME = *ALL
All defined selection conditions are to be removed.

NAME = list-poss(10): <name 1..8>
Explicit specification of selection conditions that are to be removed.

2.6.12 SAVE-SELECTED-RECORDS Output selected records

The //SAVE-SELECTED-RECORDS statement is used to save the records that have been created by editing (//START-SELECTION or //SELECT-RECORDS) in a replacement file or analysis file.

SAVE-SELECTED**-REC**ORDS
TO-REDUCTION**-NAME** = **<u>*STD</u>** / <filename 1..38 without-cat-user-gen> ,**FROM-FILE** = **<u>0</u>** / <integer 0..9> ,**ERASE-INP**UT**-FILES** = **<u>*NO</u>** / ***Y**ES

TO-REDUCTION-NAME =
This specifies whether a replacement file or an analysis file is to be created.

TO-REDUCTION-NAME = <u>*STD</u>
A replacement file with a standard file name is to be created under the default catalog ID (DEFAULT-CATID) of the SYSAUDIT user ID.
Replacement files can only be created if SATUT is running under the SYSAUDIT user ID.

TO-REDUCTION-NAME = <filename 1..38 without-cat-user-gen>
An analysis file is to be created on the current user ID.

FROM-FILE = <u>0</u> / <integer 0..9>
The work file whose contents are to be saved.

ERASE-INPUT-FILES = <u>*NO</u> / *YES
This defines whether input files are to be deleted.

ERASE-INPUT-FILES = <u>*NO</u>
The input files will not be deleted.

ERASE-INPUT-FILES = *YES
This may be specified only if a replacement file is created with TO-REDUCTION-NAME=*STD and SATUT is running under the SYSAUDIT user ID.
The input files of the SATUT session area deleted.

Exceptions:
CONSLOG files and input files that have been selected with
`//SELECT-INPUT-FILES ..., STATUS=NOT-CLOSED` are not deleted.

Notes

1. If the replacement/analysis file to be created already exists, SAT file management or SAT file evaluation has to decide in interactive mode whether or not the file is to be overwritten.
   In a batch job, execution of the statement is aborted in such an event.
2. Replacement files cannot be derived from input files with nonstandard names (analysis files), since standard input file names are required in order to compose valid replacement file names. In the event of input files with nonstandard names, an appropriate error message is returned and statement execution is aborted.

Example

SAT file management wishes to replace a number of SATLOG files by a single replacement file. This is done by first defining the input files by means of the//SELECT-INPUT-FILES statement, then specifying selection to work file 0 (//START-SELECTION) and finally issuing the following statement:

```
//save-selected-records from-file=0, to-reduction-name=*std, -
// erase-input-files=*yes
```

or in abbreviated form:

```
//save-sel-rec erase-input-files=*y
```

This statement has the effect that work file 0 is saved as the replacement file with a standard file name.

2.6.13 SELECT-INPUT-FILES Define input files

The //SELECT-INPUT-FILES statement is used to select the files that are to be selected. This statement may be specified only once per SATUT run.

```
SELECT-INPUT-FILES

INPUT-FILES = *STD(...) / list-poss(100): <filename 1..54>

    *STD(...)

        |  TYPE = list-poss(3): *SAT / *CONSLOG

        |  ,STATUS = *CLOSED / *NOT-CLOSED / *ALL

        |  ,PUBSET = *STD / list-poss(20): <cat-id 1..4>

        |  ,DATE = *ALL / <date 8..10> / *INTERVAL(...)

        |      *INTERVAL(...)

        |          |  FIRST-DATE = <date 8..10>

        |          |  ,LAST-DATE = <date 8..10>

        |  ,SESSION-NUMBER = *ALL / <integer 1..999> / *RANGE(...)

        |      *RANGE(...)

        |          |  FIRST-SESSION-NUMBER = <integer 1..999>

        |          |  ,LAST-SESSION-NUMBER = <integer 1..999>

        |  ,SEQUENCE-NUMBER = *ALL / <integer 1..999> / *RANGE(...)

        |      *RANGE(...)

        |          |  FIRST-SEQU-NUMBER = <integer 1..999>

        |          |  ,LAST-SEQU-NUMBER = <integer 1..999>
```

INPUT-FILES =
This defines the type of file to be used as input for preparation.

INPUT-FILES = *STD(...)
This means that only files with standard names (SATLOG files, replacement files and/or CONSLOG files) will be specified as input files for SATUT.

TYPE = list-poss(3): *SAT / *CONSLOG
This defines the type of file to be selected for editing.

TYPE = *SAT
Input files may be SATLOG files or replacement files only.

TYPE = *CONSLOG
CONSLOG files are used as input files.

STATUS =
This specifies the status of the input file.

STATUS = *CLOSED
The specified file must be closed.

STATUS = *NOT-CLOSED
The specified file must not be closed.

STATUS = *ALL
Any status is permissible for the specified file.

PUBSET =
This specifies the pubset on which SATUT is to look for the specified files.
For reasons of compatibility, the specification PUBLIC-VOLUME-SET or PUB-VOL-SET is still permitted in place of PUBSET.

PUBSET = *STD
SATUT is to look for the specified files on the default pubset of the SYSAUDIT user ID.

PUBSET = list-poss(20): <cat-id 1..4>
SATUT is to look for the specified files on the specified pubsets.

DATE =
This specifies the date that the input files are to have.

DATE = *ALL
Any date is permissible in the standard name of the input file.

DATE = <date 8..10>
Creation date as specified in the standard name of a SATLOG file or a CONSLOG file. In the case of a replacement file, the creation date is identical with that of the first SAT logging file with standard name used for its creation.

DATE = *INTERVAL(...)
The creation date in the standard input file name must be within the specified interval. The value specified for the year must lie between 1960 and 2059 and may be specified as two or four digits. A year number specified with two digits which is less than 60 is regarded as being in this century, while two-digit year numbers greater than or equal to 60 are regarded as being in the previous century. These limits must also be observed even if the year number is specified with four digits; “1955”, for example, is an invalid specification.

FIRST-DATE = <date 8..10>
Lower limit of the interval within which the creation date in the standard input file name is to lie.

LAST-DATE = <date 8..10>
Upper limit of the range within which the session number in the standard input file name is to lie.

SESSION-NUMBER =
This specifies the session number of the input files.

SESSION-NUMBER = *ALL
Any session number is permissible in the standard name of the input file.

SESSION-NUMBER = <integer 1..999>
The session number as specified in the standard name of the file. In the case of a replacement file, the session number is identical with that of the first input file with a standard name used for its creation.

SESSION-NUMBER = *RANGE(...)
The session number as specified in the standard name of the input files must be within the specified range.

FIRST-SESSION-NUMBER = <integer 1..999>
Lower limit of the range within which the session number in the standard input file name is to lie.

LAST-SESSION-NUMBER = <integer 1..999>
Upper limit of the range within which the session number in the standard input file name is to lie.

SEQUENCE-NUMBER =
This specifies the sequence number of the input files.

SEQUENCE-NUMBER = *ALL
Any sequence number is permissible in the standard name of the input file.

SEQUENCE-NUMBER = <integer 1..999>

Sequence number as specified in the standard name of the file. In the case of a replacement file, the sequence number is identical with that of the first input file with a standard name used for its creation.

SEQUENCE-NUMBER = *RANGE(...)
The sequence number as specified in the standard name of the input files must lie within the specified range.

FIRST-SEQU-NUMBER = <integer 1..999>
Lower limit of the range within which the sequence number in the standard input file name is to lie.

LAST-SEQU-NUMBER = <integer 1..999>
Upper limit of the range within which the sequence number in the standard input file name is to lie.

INPUT-FILES = list-poss(100): <filename 1..54>
File name of the analysis file to be used as the input file for SATUT.

Here you can only specify analysis files. These are files that were generated using the statement below in an earlier SATUT run:

//SAVE-SELECTED-RECORDS ..., TO-REDUCTION-NAME=<filename 1..54>

SATLOG files, replacement-files and CONSLOG files files must be declared as input files using INPUT-FILE=*STD(...).

For example, for the assignment of the file SYS.SATLOG.2017-04-24.006.02, the following INPUT-FILES specification is necessary:

```
INPUT-FILES=*STD(DATE=2017-04-24,SESSION-NUMBER=6,SEQUENCE-NUMBER=2)
```

Notes

SATUT takes the following actions to prevent possible overlappings when using SATLOG files and/or replacement files and/or CONSLOG files as input files:

1. A list of audit file names is compiled from the input file names as follows:
   - If the input file is a SATLOG file or a CONSLOG file, its name is copied.
   - If the input file is a replacement file, the names of the SATLOG files / CONSLOG files from which it was derived are copied.
2. The statement is rejected if the name of a SATLOG file / CONSLOG file appears more than once in the list compiled as described under 1.

Example

A user with the privilege SAT-FILE-MANAGEMENT or SAT-FILE-EVALUATION wishes to edit the SATLOG file of session 003 for the period 02-05-2017 to 09-05-2017:

```
//select-input-files input-files=*std(type=*sat, pubset=*std, -
//     date=*interval(first-date=2017-05-02, last-date=2017-05-09), -
//     session-number=003, sequence-number=*all)
```

or its abbreviated form:

```
//sel-inp-files *std(date=int(2004-05-02,2004-05-09),sess-num=003)
```

This statement causes SATUT to search for SAT/SATUT files (SATLOG files or replacement files) under SYSAUDIT with a creation date within the period from 02-05-2017 to 09-05-2017 and with the session number 003.

2.6.14 SELECT-RECORDS Define editing condition

The //SELECT-RECORDS statement is used to define an editing condition and output the records that match the editing condition.

The records selected with //SELECT-RECORDS are always selected from the input files and placed in work file 0.

Only one editing condition can be defined for any one evaluation run. If further conditions are to be applied to the same input files, these must be defined by means of //SELECT-RECORDS in a subsequent SATUT run.

SELECT**-REC**ORDS
CONDITION = ***NONE** / <cmd-rest 0..1800>

Notes

1. This statement continues to be supported for compatibility reasons only.
2. The functionality of //SELECT-RECORDS has been taken over by the //ADD-SELECTION-CONDITIONS and //START-SELECTION statements.

2.6.15 SHOW-REDUCTION-FILES-ORIGIN Show origin of replacement files

The //SHOW-REDUCTION-FILES-ORIGIN statement is used to output information about the files from which replacement files originate.

```
SHOW-REDUCTION-FILES-ORIGIN

PUBSET = *STD / list-poss(20): <cat-id 1..4>

,DATE = *ALL / <date 8..10> / *INTERVAL(...)

    *INTERVAL(...)

        | FIRST-DATE = <date 8..10>

        | ,LAST-DATE = <date 8..10>

,SESSION-NUMBER = *ALL / <integer 1..999> / *RANGE(...)

    *RANGE(...)

        | FIRST-SESSION-NUMBER = <integer 1..999>

        | ,LAST-SESSION-NUMBER = <integer 1..999>

,SEQUENCE-NUMBER = *ALL / <integer 1..999> / *RANGE(...)

    *RANGE(...)

        | FIRST-SEQU-NUMBER = <integer 1..999>

        | ,LAST-SEQU-NUMBER = <integer 1..999>

,OUTPUT = *SYSOUT / *SYSLST(...)

    *SYSLST(...)

        | LINES-PER-PAGE = 64 / <integer 20..255>
```

PUBSET =
This specifies the pubset on which the replacement files are to be found.
For reasons of compatibility, the specification PUBLIC-VOLUME-SET or PUB-VOL-SET is still permitted in place of PUBSET.

PUBSET = *STD
SATUT searches for the replacement files under the default catalog ID of the SYSAUDIT user ID.

PUBSET = list-poss(20): <cat-id 1..4>
SATUT searches for the replacement files under the specified catalog IDs.

DATE =
Date in the standard replacement file name.

DATE = *ALL
Any date is permissible for the selection.

DATE = <date 8..10>
Creation date of the first file (SAT or CONSLOG file) from which the replacement files were derived.

DATE = *INTERVAL(...)
The date in the standard replacement file name must lie within the specified interval. The value specified for the year must lie between 1960 and 2059 and may be specified as two or four digits. A year number specified with two digits which is less than 60 is regarded as being in this century, while two-digit year numbers greater than or equal to 60 are regarded as being in the previous century. These limits must also be observed even if the year number is specified with four digits; “1955”, for example, is an invalid specification.

FIRST-DATE = <date 8..10>
Lower limit of the interval within which the creation date in the standard replacement file name is to lie.

LAST-DATE = <date 8..10>
Upper limit of the interval within which the creation date in the standard replacement file name is to lie.

SESSION-NUMBER =
Session number in the standard replacement file name.

SESSION-NUMBER = *ALL
Any session number is permissible for the selection.

SESSION-NUMBER = <integer 1..999>
Session number of the first file (SAT or CONSLOG file) from which the replacement files were derived.

SESSION-NUMBER = *RANGE(...)
The session number in the standard replacement file name must be within the specified range.

FIRST-SESSION-NUMBER = <integer 1..999>
Lower limit of the range within which the session number in the standard replacement file name is to lie.

LAST-SESSION-NUMBER = <integer 1..999>
Upper limit of the range within which the session number in the standard replacement file name is to lie.

SEQUENCE-NUMBER =
Sequence number in the standard replacement file name.

SEQUENCE-NUMBER = *ALL
Any sequence number is permissible for the selection.

SEQUENCE-NUMBER = <integer 1..999>

Sequence number of the first file (SAT file or CONSLOG file) from which the replacement files were derived.

SEQUENCE-NUMBER = *RANGE(...)
The sequence number in the standard replacement file name must be within the specified range.

FIRST-SEQU-NUMBER = <integer 1..999>
Lower limit of the range within which the sequence number in the standard replacement file name is to lie.

LAST-SEQU-NUMBER = <integer 1..999>
Upper limit of the range within which the sequence number in the standard replacement file name is to lie.

OUTPUT = *SYSOUT
The requested information is to be output to SYSOUT.

OUTPUT = *SYSLST(...)
The requested information is to be output to SYSLST.

LINES-PER-PAGE = 64 / <integer 20..255>
This defines the number of lines on an output page.

Example

The origin of the replacement files dated 4.5.2017 is to be displayed:

```
//show-reduction-files-origin pubset=*std,date=2017-05-04, -
// output=sysout
```

or in abbreviated form:

```
//show-red-files-orig date=2017-05-04
```

Output produced by the statement

```
REDUCTION FILE NAME : :J:$SYSAUDIT.SYS.SATUT.2004-05-04.003.001
CREATION DATE:        2017-05-04-17.09.53.23
INPUT FILES:          SYS.SATLOG.2017-05-04.003.01
                      SYS.SATLOG.2017-05-04.003.02
SELECTION CONDITION : USER-ID IN-LIST('VALERIE','VALERE','MICHELE',
                      'ISABELLE','ARMAND','THIERRY','PHILIPPE') AND
                      FILNAME IN-LIST ('SYS.SATUT.2017-05-01.001.001',
                                       'SYS.SATUT.2017-05-01.001.002')
```

2.6.16 SHOW-SELECTED-RECORDS Print selected records

The //SHOW-SELECTED-RECORDS statement is used to request output to SYSLST or an XML file of the records previously selected by means of the //START-SELECTION statement.

The records can be output either in full or in part, with only a specified part of their contents. A sort criterion may be used to arrange the records for output according to items of information that are always present.

SHOW-SELECTED-**REC**ORDS

INFORMATION = ***ALL-F** IELDS / list-poss(100): <structured-name>

, **SORT-CRIT**ERION = ***NONE** / ***USER-ID** / ***TSN** / ***EVT** / ***TIME**STP

, **FROM-FILE** = **0** / <integer 0..9>

, **OUTPUT** = ***SYSLST** (...)

***SYSLST** (...)

| **LINE**S-**P**ER-**PAGE** = **64** / <integer 20..255>

, **XML-OUTPUT** = ***NONE** / ***STD** / <filename 1..38 without-cat-gen-user>

INFORMATION =
This defines which information from the records is to be output.
EVT and TIMESTP and - if present - RES, TSN and USER-ID are always output for all records, irrespective of the values specified for the INFORMATION operand.

INFORMATION = *ALL-FIELDS
All records are to be output in full.

INFORMATION = list-poss(100):<structured-name>
This defines the field names of the information from the records whose contents are to be output (see “Table of auditable information (field names)”).

Since TIMESTP, TSN, USER-ID, RES and EVT are always output for all records, they must not be specified here.

SORT-CRITERION =
This defines the sort criterion for output of the records.

If any sort criterion other than NONE is chosen, SAT requires work files in order to execute the sort operation.

SORT-CRITERION = *NONE
The records are not to be sorted according to a specific criterion.

SORT-CRITERION = *TIMESTP
The records are to be sorted by their time stamp.

SORT-CRITERION = *TSN
The records are to be sorted by TSN and time stamp.

SORT-CRITERION = *USER-ID
The records are to be sorted by user ID and time stamp.

SORT-CRITERION = *EVT
The records are to be sorted by event type and time stamp.

FROM-FILE = 0 / <integer 0..9>
Work file whose contents are to be output.

OUTPUT = *SYSLST(...)
The information is output to SYSLST.

LINES-PER-PAGE = 64 / <integer 20..255>
This defines the number of lines on an output page.

XML-OUTPUT =
Specifies whether the information is to be output to an XML file.

XML-OUTPUT = *NONE
No XML file is generated.

XML-OUTPUT = *STD
The information is output to a file in XML format. This file is created with the default name $SYSAUDIT.SYS.SATUT.yyyy-mm-dd.sss.nnn.XML, where:

yyyy-mm-dd	Creation date of the (temporally) first file of the input files from which the XML was generated
sss	Session number
nnn	Sequence number of the file in the session (1..999)

XML-OUTPUT = <filename 1..38 without-cat-gen-user>
The information is output in XML format to a file with the specified name.

If the file already exists, the user receives an inquiry in dialog mode asking whether the file should be overwritten. In a batch job the command is rejected and a message to this effect is issued.

Notes

1. The contents of the work files (0 to 9) are not modified by this statement.
2. The entries in CONSLOG files are converted into corresponding SATLOG records of the CLG or SKP type, as appropriate, before they are output.
3. The filpos, curlim2 and maxlim2 fields are always output in the unit 512B (= multiple of 512 bytes).

Example

The contents of work file 0 are to be output to SYSLST. The time stamp is to be used as the sort criterion:

```
//show-selected-records sort-criterion=*timestmp, from-file=0, -
// output=*syslst
```

During generation of the SATLOG file, which forms the basis for this evaluation, LOGGING-QUANTITY=*EXTENDED was activated.

Output format

```
SATUT                  V05.5A                        2018-03-30 12:18:
11                                                 PAGE       1
 PROCESSED  STATEMENT : SHOW-SELECTED-RECORDS

*****************************************************************************************************************
```

The output for the UAD event (“Add user ID”) contains a parameter list, in this case the list for the ADD-USER command. The character string “*SECRET“ in this output is entered by SATCP in the SATLOG record in place of the password which is contained in this parameter list.

Evaluation notes for *LNG fields in SATLOG records

In order to be able to recognize unauthorized intervention in computer center operation even better, for certain events the contents of the parameter list via which the event was initiated are additionally recorded. The recording only takes place if the operand LOGGING-QUANTITY=*EXTENDED was specified for the preselection in SATCP. Since parameter lists can generally be longer than 255 bytes, they are recorded in the form of *LNG fields (see section “Structure of the SATLOG files”).

*LNG fields are edited in hexadecimal and character form by SATUT. The corresponding interface macros (MF=D) must generally be used for evaluation of the parameter list. In the relevant field description, SATUT provides brief information on the basis of which the contents of the parameter list can be evaluated.

If a macro is specified for an interface description, then the macro will normally be contained in the library $TSOS.MACROLIB.

If the contents of a parameter list exceed the capacity of a SATLOG record, the parameter list is split over several SATLOG records. All the subrecords of such a SATLOG record contain the same information in their fixed part, so evaluation of each subrecord by SATUT is possible.

As a result of asynchronous processing in SATCP, the sequence of the subrecords in the SATLOG file is not guaranteed. When evaluation by SATUT is performed, sorting may be necessary. SATUT indicates in the field description which part of the field is listed. The last component part of a SATLOG record is identified by the character string “LAST” instead of a number. In addition, the displacement from the beginning of the parameter list is output.

Example

The following output contains a parameter list which has been split over two SATLOG subrecords. In the first subrecord (LOG_REC_PART = 1) the first 928 bytes of the parameter list are displayed (displacement 0000 through 039F), and in the second and last part the remaining 342 bytes (displacement 03A0 through 04F5).

```
SATUT                  V05.5A                        2018-03-09 13:47:
06                                                 PAGE       1
PROCESSED  STATEMENT : SHOW-SELECTED-RECORDS
*****************************************************************************************************************
```

2.6.17 SHOW-SELECTION-CONDITIONS Show selection conditions

The SHOW-SELECTION-CONDITIONS statement is used to output information about the selection conditions to SYSOUT or SYSLST.

SHOW-SELECTION-**COND**ITIONS

NAME = ***ALL** / list-poss(10): <name 1..8>

, OUTPUT = ***SYSOUT** / ***SYSLST**(...)

***SYSLST**(...)

| LINES-PER-PAGE = **64** / <integer 20..255>

NAME = *ALL / list-poss(10): <name 1..8>
The name of the selection conditions.

NAME = *ALL
All selection conditions are to be output.

NAME = list-poss(10): <name 1..8>
The name of the selection condition whose value is to be output.

OUTPUT = *SYSOUT / *SYSLST(...)
This specifies the destination of the requested information.

OUTPUT = *SYSOUT
The requested information is to be output to SYSOUT.

OUTPUT = *SYSLST(...)
The requested information is to be output to SYSLST.

LINES-PER-PAGE = 64 / <integer 20..255>
This specifies the output format. The default value is 64.

Example

The value of condition COND1 is to be output:

```
//show-selection-conditions name=cond1
```

Output produced by the statement

```
SELECTION CONDITION NAME : COND1
SELECTION CONDITION      :
                           EVT EQUAL ‘FRD‘
```

2.6.18 SHOW-STATISTICS Output SAT statistics

The //SHOW-STATISTICS statement is used to output selected records in the form of statistics. These are selected records from the SATUT input files or records that have been selected and edited by means of //START-SELECTION.

The available information is interpreted according to the selected input. SAT regards all data provided as input as belonging to a single session. The values for “begin of analyzed period” and “end of analyzed period” are taken from the first entry in the first file and the last entry in the last file, respectively.
The elapsed time is the difference between these two dates. If these two dates are very far apart, and if there are large gaps in the data provided as input, the results can be unexpected. Any evaluation executed by SAT must therefore be interpreted according to the selected data.

The data provides the following information:

1. global SAT logging statistics
2. a summary of the all events in event classes
3. complete SAT statistics for each event type
4. optionally, a histogram showing the number of events per minute can be output to SYSLST

```
SHOW-STATISTICS

MACHINE-SPEED = *UNDEFINED / <fixed 0.01..500>

, FROM-FILE = 0 / <integer 0..9> / *INPUT-FILES

, HISTOGRAM = *NO / *YES

, OUTPUT = *SYSLST(...) / SYSOUT

    *SYSLST (...)

        |    LINES-PER-PAGE = 64 / <integer 20..255>
```

MACHINE-SPEED = <fixed 0.01..500>
The speed of the CPU in RPF. This parameter is used for calculation of the number of records per RPF and hour in the SAT statistics.

If no value is specified for this oparand, in the output of statistics the following elements are dropped:

- the line "Machine speed" in the global SAT data

  and

- the column "# events/Mips/h" in the summary of events.

FROM-FILE = 0 / <integer 0..9> / *INPUT-FILES
The file from which the statistics are to be generated.

FROM-FILE = 0 / <integer 0..9>
The statistics are to be generated from the work file with the specified number.

FROM-FILE = *INPUT-FILES
The statistics are to be generated from the SATUT input files (//SELECT-INPUT-FILES).

HISTOGRAM =
This specifies whether or not a histogram is to be output. A histogram can be output only if OUTPUT=*SYSLST is specified.

HISTOGRAM = *NO
No histogram is to be generated and output.

HISTOGRAM = *YES
A histogram is to be generated and output.

OUTPUT = *SYSLST(...) / *SYSOUT
This specifies where the requested information is to be output.

OUTPUT = *SYSLST(...)
The requested information is to be output to SYSLST.

LINES-PER-PAGE = 64 / <integer 20..255>
This defines the number of lines per output page.

OUTPUT = *SYSOUT
The requested information is to be output to SYSOUT.

Notes

1. This statement evaluates all specified files sequentially by time. If the SAT logging has been deactivated in the meantime, or if no events occurred for a period, e.g. because of low system utilization, then this has the following effects. For the first four minutes of an “eventless” period SATUT outputs histogram lines containing null values. In order that the list does not contain an unnecessarily high number of blank lines, from the 5th minute of the “eventless” period onward SATUT suppresses the output of blank
histogram lines until such time as the next event occurs. The period during which the output was interrupted is represented by means of a line with the following contents:

   ```
   *** ----------- No events for n minutes ------------ ***
   ```

   where n is the number of minutes for which the output was suppressed.

   Example

   Extract from a histogram containing periods when no events occurred

```
2017/04/02  12:20         741 |FFFFFFFFF|FFFFFFFFF|LLLLLLLLL|LLLLLSXXX|XXXX
2017/04/02  12:21         623 |FFFFFFFFF|FFFFFFFFF|FFFFFFFFF|FFFFFFLSX
2017/04/02  12:22         217 |FFFFFFFFF|FFF
2017/04/02  12:23           0 |
2017/04/02  12:24           0 |
2017/04/02  12:25           0 |
2017/04/02  12:26           0 |
*** ----------- No events for      2 minutes ------------ ***
2017/04/02  12:29           7 |BJSX
2017/04/02  12:30           0 |
2017/04/02  12:31          18 |BFJLUX
2017/04/02  12:32           0 |
2017/04/02  12:33           7 |BFJ
2017/04/02  12:34           0 |
2017/04/02  12:35           0 |
2017/04/02  12:36           0 |
2017/04/02  12:37           0 |
*** ----------- No events for     25 minutes ------------ ***
2017/04/02  13:03           4 |SX
```

In this example there are two periods without any logged events (from 12:23 to

12:28 and from 12:34 to 13:02). The first four minutes of these periods are each represented by histogram lines containing null values, and the remaining time by the output of a line indicating the length of the period in minutes.

2. The contents of the work files (0 to 9) are not modified with this statement.
3. If the CONSLOG files do not have the same format as the SAT files, they must be converted before they can be output in statistics. If the files that are to be output in statistics have not yet been edited, message SAE5152 is output and execution of the statement is aborted.

Output of the statistics

The result of the //SHOW-STATISTICS statement is output to SYSLST or SYSOUT. This output consists of a number of tables, which are explained below.

1. Global SAT data

- The date and time of day of the evaluation run
- The names of the input files or the names of the files specified in //SELECT-RECORDS or //START-SELECTION
- The start and end of the evaluated period
  The start time and date of the SATUT run are taken from the time stamp of the first record of the oldest SAT file available. The end time and date are taken from the time stamp of the last record of the most recent SAT file.
- Elapsed time: difference between the start and end time in seconds
- Machine speed: computer power in MIPS if the MACHINE-SPEED operand has been specified
- Records/hour: average number of records logged in the SAT file(s) per hour
- # of records: Total number of audit records of the SAT input file(s) or the total number of records selected by means of //SELECT-RECORDS or //START-SELECTION
- Mean length: average length of the audit records, based on all the selected audit records from the SAT input files

- Mean kbytes/hour: Average number of bytes written to the SAT input file(s) in one hour. This value is meaningless if only individual records were selected by means of //SELECT-RECORDS or //START-SELECTION.

Example

`//show-statistics output=*sysout`

```
Input-files of statement = :PCO4:$SYSAUDIT.SYS.SATLOG.2017-04-01.137.01
                           :PCO4:$SYSAUDIT.SYS.SATLOG.2017-04-01.137.02
                           :PCO4:$SYSAUDIT.SYS.SATLOG.2017-04-02.137.03
Begin of analyzed period : 2017/04/01  10:19:21.24
End   of analyzed period : 2017/04/02  17:44:34.23
Elapsed time             =      113113 s     =     1 d  26713 s
Records/hour             =       53.56
# of records             =        1683
Mean length              =      102.00
Mean kbytes/hour         =        5.54
```

2. Summary of the most important events

All events are collected in event classes in this statistic.

In all cases it is the total number of events that have occurred in each event class that are calculated (# events), as well as the average number of these events per hour (# events/h) and, if the MACHINE-SPEED operand has been specified, the average number of these events per MIPS per hour (# events/Mips/h).

Statistics are collected for the following event classes:

1. DMS
   - Files (file accesses)
   - Security (access to protection attributes in the user catalog)
   - Rename Files (renaming of files)
2. Catalog Management
   (import and export of pubsets)
3. Job Enable (Dialog & Batch)
   (LOGON for interactive and batch tasks, both successful and unsuccessful)
4. Job (Rest)
   (job events without LOGON for interactive and batch jobs)
5. Job Variables
   (JVs: accesses to job variables)
6. BLS
   (Binder and Loader System: loading and unloading of (parts of) TU programs)
7. Spool
   - Jobs (commands)
   - Devices (administration)

8. PLAM/LAM
   (access to library members)
9. DSSM (Dynamic SubSystem Management)
   - Connection/Disconnection:connection to/disconnection from privileged subsystems of TU jobs)
   - Catalog Management (DSSM catalog management)
10. Syntax Files
    (access to syntax files)
11. Users/Groups/Privileges
    - Users (user ID administration)
    - Privileges (privileges management)
    - Groups (user group administration)
12. Object Protection
    - GUARDS (Events relating to the corresponding objects)
    - Coowner Protection (Events relating to the corresponding objects)
    - Default Protection (Events relating to the corresponding objects)
    - Access Control List (ACL: accesses to ACL entries)
13. System Access Control Management
    - Terminal Sets (administration of terminal sets)
    - Operator Roles (administration of operator roles)
    - Keys (object KEY)
14. SAT

    (SAT events)
15. UTM

    (UTM events)
16. SESAM

    (SESAM events)
17. POSIX
    - Files and Directories
    - Child Processes
    - Processes
    - System Resources

    (Events relating to the corresponding objects)
18. Communication Methods
    - DCAM
    - BCAM
    - IP Security

    (Events relating to the corresponding objects)
19. Memory Pools

    (opening, closing, releasing, accessing)

20. Events
    - Serialization
    - Eventing

    (activation and deactivation of eventing and serialization)
21. Fast Intertask Communication

    (object FITC)
22. Storage Class Events

    (object SMS)
23. Data Spaces
24. Volume

    (processing and management of tapes using MAREN; volume processing with NDM; volume processing with FDDRL, VOLIN or IOCFCOPY)
25. ADAM Device Management
26. ANY event (system exit)

Example

```
SUMMARY OF EVENTS
-----------------
     Event class                        # events  # events/h  # events/Mips/h
     -----------                        --------  ----------  ---------------
 1 : DMS
       Files                                 711       22.63            4.53
       Security                              733       23.33            4.67
       Rename Files                            5        0.16            0.03
 2 : Catalog Management                        0        0.00            0.00
 3 : Job Enable (Dialog & Batch)
       Success                                33        1.05            0.21
       Failure                                 6        0.19            0.04
 4 : Job (Rest)                                33        1.05            0.21
 5 : Job Variables                              0        0.00            0.00
 6 : BLS                                       15        0.48            0.10
 7 : Spool
       Jobs                                    0        0.00            0.00
       Devices                                 0        0.00            0.00
 8 : PLAM/ILAM                                 20        0.64            0.13
 9 : DSSM
       Connection/Disconnection                2        0.06            0.01
       Catalog Management                     17        0.54            0.11
10 : Syntax Files                              0        0.00            0.00
11 : Users/Groups/Privileges
       Users                                  87        2.77            0.55
       Privileges                              0        0.00            0.00
       Groups                                  0        0.00            0.00
12 : Object Protection
       GUARDS                                  0        0.00            0.00
       Coowner Protection                      0        0.00            0.00
       Default Protection                      0        0.00            0.00
       Access Control List                     0        0.00            0.00
13 : System Access Control Management
       Terminal Sets                           0        0.00            0.00
       Operator Roles                          0        0.00            0.00
       Keys                                   13        0.41            0.08
14 : SAT                                       2        0.06            0.01
15 : UTM                                       0        0.00            0.00
16 : SESAM                                     0        0.00            0.00
17 : POSIX
       Files and Directories                   0        0.00            0.00
       Child Processes                         0        0.00            0.00
       Processes                               0        0.00            0.00
       System Resources                        0        0.00            0.00
18 : Communication Methods
       DCAM                                    0        0.00            0.00
       BCAM                                    0        0.00            0.00
       IP Security                             0        0.00            0.00
19 : Memory Pools                              0        0.00            0.00
20 : Events
       Serialization                           0        0.00            0.00
       Eventing                                0        0.00            0.00
21 : Fast Intertask Communication              0        0.00            0.00
22 : Storage Class Events                      0        0.00            0.00
23 : Data Spaces                               0        0.00            0.00
24 : Volume                                    0        0.00            0.00
25 : ADAM device management                    0        0.00            0.00
26 : ANY event (system exit)                   0        0.00            0.00
```

3. Complete SAT statistics for each event type

The statistical data for each individual event is listed in alphabetical order after the corresponding abbreviation (see section "Table of object-related events").

The following information is supplied for each event:

EVENT	*Abbreviation denoting the event*
# SUCC	*Number of audit records with the result SUCCESS logged for this event*
# FAIL	*Number of audit records with the result FAILURE logged for this event*
# NONE	*This number should be 0; if it is not, you should check whether these audit records might not have come from an exit routine of the computer center (event type ANY); if this is not the case, you should contact the maintenance personnel.)*
LEN SUCC	*Average length of the audit records for this event type which have the result SUCCESS*
LEN FAIL	*Average length of the audit records for this event type which have the result FAILURE*
LEN NONE	*Average length of the audit records for this event type without a result*
% EVENTS	*Relationship (expressed as a percentage) between the totals of #SUCC, # FAIL and # NONE and the total number of all audit records to be evaluated*
% FAIL (EVENT)	*Relationship (expressed as a percentage) between the total number of audit records of this event type with the result FAILURE and the total number of audit records with the results SUCCESS, FAILURE and NONE*
RECORDS /HOUR	*Average number of audit records for this event type which are logged in an hour*

These statistics cover only those event types which actually exist in the input files for SHOW-STATISTICS.

The values # NONE, LEN SUCC, LEN FAIL and LEN NONE are suppressed for output to SYSOUT.

Example

```
#  EVENT  # SUCC   # FAIL   # NONE    LEN SUCC   LEN FAIL   LEN NONE  % EVENTS  % FAIL(EVENT)    RECORDS
/HOUR
--- -----  ------   ------   ------    --------   --------   --------  --------  -------------
------------
  1 FCD        59        0        0      108.61       0.00       0.00      3.51           0.00
1.88
  2 FCL       298        0        0       97.52       0.00       0.00     17.71           0.00
9.48
  3 FCS        58        0        0      106.02       0.00       0.00      3.45           0.00
1.85
  4 FDD        64        2        0      100.62      99.00       0.00      3.92           3.03
2.10
  5 FDS        64       20        0      103.94     100.50       0.00      4.99          23.81
2.67
  6 FED        33        0        0      101.09       0.00       0.00      1.96           0.00
1.05
  7 FMD        92        0        0       95.16       0.00       0.00      5.47           0.00
2.93
  8 FMS        11        0        0      121.91       0.00       0.00      0.65           0.00
0.35
  9 FRD       162        1        0      100.68      97.00       0.00      9.69           0.61
5.19
 10 FRN         5        0        0      148.60       0.00       0.00      0.30           0.00
0.16
 11 FRS       391      189        0      106.45     110.13       0.00     34.46          32.59
18.46
 12 JBE        16        2        0       81.06      71.50       0.00      1.07          11.11
0.57
 13 JDE        17        4        0       79.29      83.75       0.00      1.25          19.05
0.67
 14 JED        16        0        0       58.94       0.00       0.00      0.95           0.00
0.51
 15 JIN        17        0        0       69.00       0.00       0.00      1.01           0.00
0.54
 16 KTC        13        0        0      125.00       0.00       0.00      0.77           0.00
0.41
 17 LCL        10        0        0      128.50       0.00       0.00      0.59           0.00
0.32
 18 LEE        10        0        0      124.50       0.00       0.00      0.59           0.00
0.32
 19 SCR        14        3        0       79.71      78.33       0.00      1.01          17.65
0.54
 20 SDS         2        0        0       85.00       0.00       0.00      0.12           0.00
0.06
 21 UAD         2        0        0       81.00       0.00       0.00      0.12           0.00
0.06
 22 UCK        71       10        0       82.70      82.90       0.00      4.81          12.35
2.58
 23 UML         2        0        0       81.00       0.00       0.00      0.12           0.00
0.06
 24 URM         1        1        0       81.00      81.00       0.00      0.12          50.00
0.06
 25 XLD        10        0        0      199.00       0.00       0.00      0.59           0.00
0.32
 26 XUL         5        0        0       80.00       0.00       0.00      0.30           0.00
0.16
 27 ZBG         3        0        0      245.33       0.00       0.00      0.18           0.00
0.10
 28 ZCH         2        0        0      130.00       0.00       0.00      0.12           0.00
0.06
 29 ZND         3        0        0       83.33       0.00       0.00      0.18           0.00
0.10
--- -----  ------   ------   ------    --------   --------   --------  --------  -------------
------------
TOTAL:       1451      232        0      101.26     106.65       0.00    100.00          13.78
53.6
```

4. Optional: histogram representing the number of events per minute

A histogram depicting the number and type of events is generated for each minute within the auditing period.

The axis on which the number of events is represented is subdivided into percentage fractions. Each segment represents 10%, and each letter represents 1% of the maximum possible number of events per minute. Each minute in which this value was reached is marked in the histogram by an asterisk (*).

Each line in the histogram contains the following information (from left to right):

- *date*
- *time of day*
- *number of events in this minute*
- *'*' if the maximum possible number of events per minute was logged*
- *line in the histogram showing the distribution of the events for each event type.*

Example (SYSLST)

```
/show-statistics from-file=*input-files, -
/ histogram=*yes, output=*syslst
```

Note

If there are gaps in the sequence of the SAT files specified or in the period of time which is covered by the input files, these gaps are treated like periods during which no events occurred. In the first four minutes of such a gap the histogram contains lines in which the number of events per minute is equal to zero; from the fifth minute onward a line is output indicating the length of time during which no events occurred.

```
SATUT                  V05.5A                        2017-04-06 16:17:
07                                                 PAGE     4
PROCESSED  STATEMENT : SHOW-STATISTICS
*************************************************************************************************
```

2.6.19 START-SELECTION Initiate evaluation

The //START-SELECTION statement is used to select records which satisfy a predefined selection condition. The edited records are stored in a separate work file for each selection condition; each file is identified by a number. This takes place in one step, i.e. each input file is read only once.

The output result of the statement is the number of edited records per work file.

```
START-SELECTION

  FROM-FILE = 0 / <integer 0..9> / *INPUT-FILES

  ,TO-FILE = list-poss(10): *PARAMETERS(...)

      *PARAMETERS(...)

          |    FILE = 0 / <integer 0..9>

          |    ,CONDITION-NAME = <name 1..8>
```

FROM-FILE = 0 / <integer 0..9> / *INPUT-FILES
Source of the records.

FROM-FILE = 0 / <integer 0..9>
The records are selected from the work file with the specified number.

FROM-FILE = *INPUT-FILES
The records are selected from the SATUT input files that were selected in the SELECT-INPUT-FILES statement.

TO-FILE = *PARAMETERS(...)
This specifies the condition and the output destination after editing.

FILE = 0 / <integer 0..9>
This specifies the work file to which the selected records are to be written.

CONDITION-NAME = <name 1..8>
The name of the selection condition that was specified in the //ADD-SELECTION-CONDITIONS statement.

Notes

1. *If a work file that is specified for output in the TO-FILE operand has already been used in a SATUT run, it will be overwritten.*
2. *If the same file is specified in the FROM-FILE and TO-FILE operands, the work file will be overwritten.*

Example

Records from the input files are selected according to selection condition COND1 and are stored in work file 3.

```
//start-selection from-file=*input-files, -
// to-file=*parameters(file=3,condition=cond1)
```

The output result of the statement is the number of edited records:

```
SAE7001: START-SELECTION TERMINATED.‘123‘ RECORDS SELECTED IN WORK
FILE‘3‘
```

2.6.20 Example of evaluation

Examples of the formation of complex condition expressions are given in "ADD-SELECTION-CONDITIONS Define selection conditions".Examples of evaluation in connection with preselection and postselection are provided in "Monitoring special security-relevant activities".

In this example, the SAT file manager would like to achieve the following:

1. Detect potential attempts at intrusion during the preceding session. To do this it is necessary to select the audit records of rejected LOGON attempts from the SATLOG file.
2. Create an analysis file containing all events that relate to file objects. This file is to be analyzed decentrally at a later time.

Prerequisites

- The audit attribute for all switchable user IDs, in other words all except those with the SECURITY-ADMINISTRATION or FILE-MANAGEMENT privilege, has been set to OFF:

  /modify-sat-preselection user-auditing=*all-switchable(*off)
- The audit attribute of all events for which it is allowed to be changed has been set to OFF in the preselection (see “Individual control of selection). Exception: if the “check user ID” event (UCK) with the result “FAILURE” has been selected for logging:

  /modify-sat-preselection event-auditing=
  uck(audit-switch=*on(result=*failure))
- The session to be evaluated was session number 137, beginning on 2018-03-01.

The SAT file manager begins evaluation by starting SATUT:

/ start-satut

The input file that is selected is the SATLOG file from session 137:

//select-input-files input-files=*std(session-number=137)

To obtain an overview of the activities in the selected session, the SAT file manager arranges for statistics to be output to SYSOUT:

//show-statistics output=*sysout

The following output is obtained (the precise meaning of the individual output fields is explained with the statement SHOW-STATISTICS):

```
Input-files of statement = :PCO4:$SYSAUDIT.SYS.SATLOG.2018-03-01.137.01
                           :PCO4:$SYSAUDIT.SYS.SATLOG.2018-03-01.137.02
                           :PCO4:$SYSAUDIT.SYS.SATLOG.2018-03-02.137.03
Begin of analyzed period : 2018/03/01  10:19:21.24
End   of analyzed period : 2018/03/02  17:44:34.23
Elapsed time             =      113113 s     =     1 d  26713 s
Records/hour             =       53.56
# of records             =        1683
Mean length              =      102.00
Mean kbytes/hour         =        5.54
```

SUMMARY OF EVENTS

	Event-class	# events	# events/h
1 :	DMS		
	Files	711	22.63
	Security	733	23.33
	Rename Files	5	0.16
2 :	Catalog Management	0	0.00
3 :	Job Enable (Dialog & Batch)		
	Success	33	1.05
	Failure	6	0.19
4 :	Job (Rest)	33	1.05
5 :	Job Variables	0	0.00
6 :	BLS	15	0.48
7 :	Spool		
	Jobs	0	0.00
	Devices	0	0.00
8 :	PLAM/ILAM	20	0.64
9 :	DSSM		
	Connection/Disconnection	2	0.06
	Catalog Management	17	0.54
10 :	Syntax Files	0	0.00
11 :	Users/Groups/Privileges		
	Users	87	2.77
	Privileges	0	0.00
	Groups	0	0.00
12 :	Object Protection		
	GUARDS	0	0.00
	Coowner Protection	0	0.00
	Default Protection	0	0.00
	Access Control List	0	0.00
13 :	System Access Control Management		
	Terminal Sets	0	0.00
	Operator Roles	0	0.00
	Keys	13	0.41
14 :	SAT	2	0.06
15 :	UTM	0	0.00
16 :	SESAM	0	0.00
17 :	POSIX		
	Files and Directories	0	0.00
	Child Processes	0	0.00
	Processes	0	0.00
	System Resources	0	0.00
18 :	Communication Methods		
	DCAM	0	0.00
	BCAM	0	0.00
	IP Security	0	0.00
19 :	Memory Pools	0	0.00
20 :	Events		
	Serialization	0	0.00
	Eventing	0	0.00
21 :	Fast Intertask Communication	0	0.00
22 :	Storage Class Events	0	0.00
23 :	Data Spaces	0	0.00
24 :	Volume	0	0.00
25 :	ADAM device management	0	0.00
26 :	ANY event (system exit)	0	0.00

```
 #  EVENT  # SUCC   # FAIL   # NONE    LEN SUCC   LEN FAIL   LEN NONE  % EVENTS  % FAIL(EVENT)    RECORDS
/HOUR
--- -----  ------   ------   ------    --------   --------   --------  --------  -------------
------------
  1 FCD        59        0        0      108.61       0.00       0.00      3.51           0.00
1.88
  2 FCL       298        0        0       97.52       0.00       0.00     17.71           0.00
9.48
  3 FCS        58        0        0      106.02       0.00       0.00      3.45           0.00
1.85
  4 FDD        64        2        0      100.62      99.00       0.00      3.92           3.03
2.10
  5 FDS        64       20        0      103.94     100.50       0.00      4.99          23.81
2.67
  6 FED        33        0        0      101.09       0.00       0.00      1.96           0.00
1.05
  7 FMD        92        0        0       95.16       0.00       0.00      5.47           0.00
2.93
  8 FMS        11        0        0      121.91       0.00       0.00      0.65           0.00
0.35
  9 FRD       162        1        0      100.68      97.00       0.00      9.69           0.61
5.19
 10 FRN         5        0        0      148.60       0.00       0.00      0.30           0.00
0.16
 11 FRS       391      189        0      106.45     110.13       0.00     34.46          32.59
18.46
 12 JBE        16        2        0       81.06      71.50       0.00      1.07          11.11
0.57
 13 JDE        17        4        0       79.29      83.75       0.00      1.25          19.05
0.67
 14 JED        16        0        0       58.94       0.00       0.00      0.95           0.00
0.51
 15 JIN        17        0        0       69.00       0.00       0.00      1.01           0.00
0.54
 16 KTC        13        0        0      125.00       0.00       0.00      0.77           0.00
0.41
 17 LCL        10        0        0      128.50       0.00       0.00      0.59           0.00
0.32
 18 LEE        10        0        0      124.50       0.00       0.00      0.59           0.00
0.32
 19 SCR        14        3        0       79.71      78.33       0.00      1.01          17.65
0.54
 20 SDS         2        0        0       85.00       0.00       0.00      0.12           0.00
0.06
 21 UAD         2        0        0       81.00       0.00       0.00      0.12           0.00
0.06
 22 UCK        71       10        0       82.70      82.90       0.00      4.81          12.35
2.58
 23 UML         2        0        0       81.00       0.00       0.00      0.12           0.00
0.06
 24 URM         1        1        0       81.00      81.00       0.00      0.12          50.00
0.06
 25 XLD        10        0        0      199.00       0.00       0.00      0.59           0.00
0.32
 26 XUL         5        0        0       80.00       0.00       0.00      0.30           0.00
0.16
 27 ZBG         3        0        0      245.33       0.00       0.00      0.18           0.00
0.10
 28 ZCH         2        0        0      130.00       0.00       0.00      0.12           0.00
0.06
 29 ZND         3        0        0       83.33       0.00       0.00      0.18           0.00
0.10
--- -----  ------   ------   ------    --------   --------   --------  --------  -------------
------------
TOTAL:       1451      232        0      101.26     106.65       0.00    100.00          13.78
53.6
```

The SAT file manager defines the first selection condition with the name “badlog”. This condition relates to all records that concern the “check user ID” event with the result “FAILURE”.

```
//add-selection-conditions name=badlog, -
// condition=evt equal ’uck’ and res equal f
```

The second selection condition by the name of “file” relates to records in which events are logged whose short name begins with the letter “F”. These are all events which relate to file objects.

```
//add-selection-conditions name=file,condition=evt match ’f*’
```

The following command has the effect that editing for both conditions is executed in one step. All records that satisfy the selection condition “badlog” are written to work file 0, while all records that satisfy the “file” condition are written to work file 5.

```
//start-selection from-file=*input-files, -
//                  to-file=(*parameters(condition-name=badlog), -
//                           *parameters(file=5,condition-name=file))
```

```
%  SAE7001 'START-SELECTION' STATEMENT TERMINATED. '10' RECORDS SELECTED IN WORK FILE
 ' 0'
%  SAE7001 'START-SELECTION' STATEMENT TERMINATED. '1449' RECORDS SELECTED IN WORK
 FILE ' 5'
```

There were therefore 10 unsuccessful LOGON attempts and 1449 events relating to file objects.
The records with the events relating to file objects are written to the ANALYZE.FILE-EVENTS file for the purpose of decentralized analysis.

```
//save-selected-records to-reduction-name=analyze.file-events,from-file=5
```

As the SAT file manager considers the number of unsuccessful LOGON attempts to be too high for immediate evaluation, he would like to restrict the selection still further. The first step is to obtain information about the existing selection conditions.

```
//show-selection-conditions
```

```
SELECTION CONDITION NAME : BADLOG
SELECTION CONDITION      :
                           EVT EQUAL 'UCK'
                         AND RES EQUAL F
 ================================================================================
SELECTION CONDITION NAME : FILE
SELECTION CONDITION      :
                           EVT MATCH 'F*'
 ================================================================================
```

The SAT file manager would like to evaluate only failed LOGON attempts made with the “TSOS” user ID. To do that it is necessary to define another selection condition to select records containing the value TSOS in the logged data field OBJ-UID (see “Tables of auditable information on object-related events (1)”).

```
//add-selection-conditions name=uidtsos,condition=obj-uid equal ’tsos’
```

The SAT file manager then initiates a second stage of editing. All records from work file 0 that satisfy the “uidtsos” condition are to be written to work file 1. As the records in work file 0 already satisfy the “badlog” condition, the result of this editing is the set of all records for which both conditions (“badlog” and “uidtsos”) are true.

```
//start-selection from-file=0, -
//                 to-file=*parameters(file=1,condition-name=uidtsos)
```

```
%  SAE7001 'START-SELECTION' STATEMENT TERMINATED. '3' RECORDS SELECTED IN WORK FILE
 ' 1'
```

Now the result of this selection is only three records. These are to be output to SYSLST for detailed evaluation.

//show-selected-records from-file=1

Finally the SAT file manager outputs a set of statistics for the session, with a histogram, to SYSLST. The evaluation run is then terminated.

//show-statistics from-file=*input-files,histogram=*yes
//end

```
%  SAE5004 SAT FILE EVALUATOR TERMINATED NORMALLY
```

SYSLST shows the result of //SHOW-SELECTED-RECORDS on pages 1 and 2.

```
SATUT                  V05.5A                        2018-03-06 15:44:
22                                                 PAGE       1
PROCESSED  STATEMENT : SHOW-SELECTED-RECORDS
*****************************************************************************************************
```

The following part of the list shows the result of //SHOW-STATISTICS and is largely identical to the statistics output to SYSOUT at the start of the session. It also contains the histogram of the events.

```
SATUT                  V05.5A                        2018-03-06 16:17:
07                                                 PAGE     1
PROCESSED  STATEMENT : SHOW-STATISTICS
*****************************************************************************************************

EXPLANATION ON USED LETTERS:
----------------------------
F : FCD, FCL, FCS, FDD, FDS, FED, FMD, FMS, FRD, FRN, FRS
J : JBE, JDE, JED, JIN
K : KTC
L : LCL, LEE
S : SCR, SDS
U : UAD, UCK, UML, URM
X : XLD, XUL
Z : ZBG, ZCH, ZND
```

In a final evaluation stage, which can only partly be automated with SAT or by programming, the selected records must be evaluated in order to determine what further action needs to be taken, if at all.

In the example, the two selected records in SYSLST are evaluated manually on the basis of “Tables of auditable information on object-related events (1)”.

```
EVT RES DATE     TIME   TSN  USER-ID
UCK  F  20180301 163627 0DHC TSOS     OBJ-UID= TSOS                STATION= $$$06015            PROCNAM=
XYZ0231X
                                      CHKMODE= DIALOG              REJR   = 03400001
UCK  F  20180302 141855 0DHG TSOS     AUDITID= D4C3C8C88995A97CC6E2C34BD5C5E3                   OBJ-UID=
TSOS
                                      STATION= $$$06007            PROCNAM= XYZ4711X            CHKMODE=
NET-DIALOG-ACCESS
                                      REJR   = 02400001            PRINCCL= MCHHinz@FTS.NET
UCK  F  20180302 144612 0DHI TSOS     AUDITID= D4C3C8D2A495A97CC6E2C34BD5C5E3                   OBJ-UID=
TSOS
                                      STATION= $$$06009            PROCNAM= XYZ0815X            CHKMODE=
NET-DIALOG-ACCESS
                                      REJR   = 1E400001            PRINCCL= MCHKunz@FTS.NET
```

According to the table for the object USERID on "Tables of auditable information on object-related events (2)", “obj-uid” and “chkmode” are always logged, “station”, “procnam”, “rejr” and “princcl” may be logged.

One possible approach is to examine whether a cluster of logon attempts that were rejected because of user error has occurred at a particular data terminal or in a batch. This could indicate that an attempt has been made to penetrate the system by trying out different passwords.

In this case the analysis shows that only three logon attempts for TSOS (obj-uid) in dialog mode (chkmode) were rejected due to user error (rejr) throughout the entire evaluation period of more than 24 hours (see table on "Tables of auditable information on object-related events (2)"). What is more, these attempts were made from different data terminals (station and procnam). As a consequence, analysis in this case would produce the overall result “harmless”.

2.7 Table of object-related events

The table in the following shows the objects and their auditable events, the abbreviated names of the events and indication of their audit attributes.
The /MODIFY-SAT-PRESELECTION command enables the security administrator to modify the SAT preselection values for most events.

The individual columns have the following meanings:

- **OBJECT Event** columnSpecification of the object, accompanied by the operations which result in auditable events.
- **Event name** columnEach event has a 3-character event name which may be used as a keyword in the commands /SHOW-SAT-STATUS and /MODIFY-SAT-PRESELECTION as well as in the statements //ADD-SELECTION-CONDITIONS and //SELECT-RECORDS .
- **Audit attribute Chg** columnIndication of whether the audit attribute for the event can be changed.

Y (YES)	The audit attribute can be changed
N (NO)	The audit attribute cannot be changed (permanently security-relevant event)
-	Entry not relevant

- **Audit attribute Dft**

  columnShows the default setting for the audit attribute (see "Selection procedure") of the event:

A	Audit attribute ALL, i.e. the event is always logged
S	Audit attribute SUCCESS, i.e. the event is logged if it has been successfully executed (data field `res equal S` in the SATLOG record)
F	Audit attribute FAILURE, i.e. the event is logged if it has not been successfully executed (data field `res equal F` in the SATLOG record)
N	Audit attribute NONE, i.e. the event is not logged
-	Entry not relevant

Note

The events and fields documented in this manual correspond to the status at the time when the manual was published. However, the type and scope of the audited information may change for products which pass information to SAT for logging and which appear after publication of this manual. The related product manual will contain an updated list of events and fields, and you should therefore use only the information provided in these product manuals.

OBJECT Event	**Event name**	**Audit attribute**	
		Chg	**Dft**

ADAM (device management)			
Add device operation	ADO	Y	N
ANY			
Any event (system exit) (see note)	ANY	Y	N
APPLICATION (DCAM)			
Open application (YOPEN)	DON	Y	N
Close application (YCLOSE)	DCL	Y	N
Connect application (YOPNCON)	DCN	Y	N
Disconnect application (YCLSCON)	DDS	Y	N
BCAM			
Open TSAP	BAO	Y	N
Close TSAP	BAC	Y	N
Open connection	BCN	Y	N
Close connection	BDS	Y	N
CATALOG (PVS)			
Start import pubset task	CIP	Y	S
Start export pubset task	CEP	Y	S
Check catalog	CKR	Y	N
Convert catalog	CVR	Y	N
CONSLOG (see note)			
CONSLOG entry		-	-
COOWNER PROTECTION			
Add co-owner protection rule	CRA	Y	N
Modify co-owner protection rule	CRM	Y	N
Display co-owner authorization rule	CRQ	Y	N
Remove co-owner protection rule	CRR	Y	N
Display co-owner protection rule	CRS	Y	N
DATA SPACES (see note)			
Create DATA SPACE	DSB	Y	N

Connect to DATA SPACE	DSC	Y	N
Release connection to DATA SPACE	DSD	Y	N
Delete DATA SPACE	DSE	Y	N
Modify/reset DATA SPACE	DSM	Y	N
DEFAULT PROTECTION			
Define default values for protection attributes	DAA	Y	N
Modify default values for protection attributes	DAM	Y	N
Display default values for protection attributes	DAS	Y	N
Add default protection rule	DRA	Y	N
Modify default protection rule	DRM	Y	N
Display default protection attributes for object	DRQ	Y	N
Remove default protection rule	DRR	Y	N
Display default protection rule	DRS	Y	N
Add user ID for object path	DUA	Y	N
Remove user ID for object path	DUR	Y	N
Display user ID for object path	DUS	Y	N
EVENTING-ITEM (see note)			
Activate eventing	EEE	Y	N
Deactivate eventing	EDE	Y	N
Activate serialization	EES	Y	N
Deactivate serialization	EDS	Y	N
FILE (see note)			
Create file	FCD	Y	N
Read file	FRD	Y	N
Execute file (open exec)	FED	Y	N
Modify file	FMD	Y	N
Close file	FCL	Y	N
Delete file	FDD	Y	N
Rename file with ARCHIVE	FAR	Y	N

Rename file	FRN	Y	N
Define protection attributes	FCS	Y	N
Modify protection attributes	FMS	Y	N
Delete protection attributes	FDS	Y	N
Read protection attributes	FRS	Y	N
Import protection attributes	FIS	Y	N
Export protection attributes	FES	Y	N
Convert file into decrypted file	FDC	Y	N
Convert file into encrypted file	FEC	Y	N
Move file extents (SPACEOPT)	FME	Y	N
Select object for reorganization (SPACEOPT)	FSO	Y	N
FITC (Fast Intertask Comm.) (see note)			
Define port access	POA	Y	N
Define port	POB	Y	N
Connect to port	POC	Y	N
Disconnect from port	POD	Y	N
Release port	POE	Y	N
Release port access	POR	Y	N
Implicit exchange with port	POX	Y	N
GROUP (user group)			
Add	GSH	Y	A
Modify	GRM	Y	A
Remove	GMD	Y	A
Show	GAD	Y	N
GUARDS			
Generate a guard	GUB	Y	N
Copy a guard	GUC	Y	N
Delete a guard	GUD	Y	N
Change guards catalog	GUF	Y	N

Repair guards catalog	GUR	Y	N
Modify attributes	GUM	Y	N
Show attributes	GUS	Y	N
Define access conditions	GAA	Y	N
Modify access conditions	GAM	Y	N
Remove access conditions	GAR	Y	N
Show access conditions	GAS	Y	N
Interrogate access conditions	GAQ	Y	N
IPSEC			
Load IPSEC security database	ILD	Y	N
Security policy infringement during data transfer	IPV	Y	N
JOB (see note)			
Initiate batch job or subtask	JBE	Y	F
Abort job	JCN	Y	N
Initiate dialog or RLOGIN	JDE	Y	A
End job	JED	Y	N
Initialize batch job or subtask	JIN	Y	A
Modify batch job	JMD	Y	N
Generate POSIX task	JFK	Y	A
JOB VARIABLES			
Rename with ARCHIVE	JVA	Y	N
Create protection attributes	JVC	Y	N
Delete protection attributes	JVD	Y	N
Modify protection attributes	JVM	Y	F
Read data (GETJV)	JVG	Y	F
Write data (SETJV)	JVS	Y	F
Query JV	JVQ	Y	N
Rename JV	JVR	Y	N
KEY			

Add KERBEROS Encryption Type	KEA	Y	A
Delete KERBEROS Encryption Type	KED	Y	A
Add KERBEROS Principal	KPA	Y	A
Delete KERBEROS Principal	KPD	Y	A
Modify KERBEROS Principal	KPM	Y	A
KERBEROS ticket check	KTC	Y	F
Abortive attempt at a crypto password check after exceeding the maximum number of abortive attempts	KXM	Y	F
MEMORY-POOL (see note)			
Enable (ENAMP)	MEN	Y	N
Disable (DISMP)	MDS	Y	N
Release (RELMP)	MRL	Y	N
Make readable for TU (with ($)CSTMP in TPR)	MRD	Y	N
Make readable with CSTMP in TU	MAC	Y	S
OPERATOR ROLE			
Add routing code	ORA	Y	N
Create operator role	ORB	Y	N
Assign operator role	ORC	Y	N
Delete operator role from user record	ORD	Y	N
Delete operator role	ORE	Y	N
Withdraw routing code	ORR	Y	N
PLAM			
Create library member	LCE	Y	N
Modify library member	LME	Y	N
Read library member	LRE	Y	N
Execute library member	LEE	Y	N
Close library member	LCL	Y	N
Delete library member	LDE	Y	N
Rename library member	LRN	Y	N

Create security attributes	LCS	Y	N
Delete security attributes	LDS	Y	N
Modify security attributes	LMS	Y	N
POSIX-CHILD-Process (see note)			
Create new process (fork)	XFK	Y	N
Create new process on rlogin access (rfork)	XRF	Y	N
POSIX-FILE-and-Directory (see note)			
Change current directory (chdir)	XCD	Y	N
Close file (close)	XCL	Y	N
Change file access rights (chmod)	XCM	Y	N
Change file owner or group (chown)	XCO	Y	N
Create new file (creat)	XCR	Y	N
Create directory via descriptor (mkdirat)	XDA	Y	N
Duplicate file descriptor (dup)	XDP	Y	N
File control operation (fcntl)	XFC	Y	N
Change current directory via descriptor (fchdir)	XFD	Y	N
Change file access rights via descriptor (fchmod)	XFM	Y	N
Change a file’s owner or group via descriptor (fchown)	XFO	Y	N
Create a link to a file via descriptor (linkat)	XLA	Y	N
Create a link to a file (link)	XLN	Y	N
Change the owner or group of a file or link (lchown)	XLO	Y	N
Change file access rights via descriptor (fchmodat)	XMA	Y	N
Create directory (mkdir)	XMD	Y	N
Map file in virtual memory(mmap)	XMM	Y	N
Set protection attributes for file mapping in virtual memory (mprotect)	XMP	Y	N
Mount file system (mount)	XMT	Y	N
Cancel mapping of file in virtual memory (munmap)	XMU	Y	N
Open file via descriptor (openat)	XOA	Y	N

Open file (open)	XOP	Y	N
Rename file via descriptor (renameat)	XRA	Y	N
Remove directory (rmdir)	XRD	Y	N
Rename file (rename)	XRN	Y	N
Create symbolic link to a file via descriptor (symlinkat)	XSA	Y	N
Create symbolic link to a file (symlink)	XSL	Y	N
Delete file or directory via descriptor (unlinkat)	XUA	Y	N
Set file bit mask for a process (umask)	XUM	Y	N
Delete file (remove/unlink)	XUN	Y	N
Unmount file system (umount)	XUT	Y	N
Change file group or owner via descriptor (fchownat)	XWA	Y	N
POSIX-PROCESS (see note)			
Set effective group number for a process (setegid)	XEG	Y	N
Set effective user number for a process (seteuid)	XEU	Y	N
Execute file (exec)	XEX	Y	N
Set maximum number of group members for a process (setgroups)	XGR	Y	N
Send signal to process or process group (kill)	XKL	Y	N
Set process limits (ulimit)	XLM	Y	N
Set real and effective group number for a process (setregid)	XRG	Y	N
Set real and effective user number for a process (setreuid)	XRU	Y	N
Set group number of a process (setgid)	XSG	Y	N
Set process group number (setpgrp)	XSP	Y	N
Set limit value for a resource (setrlimit)	XSR	Y	N
Set user number of a process (setuid)	XSU	Y	N
POSIX-SYSTEM-Resources (see note)			
Change system time (adjtime)	XAJ	Y	N
Set user attributes (pwent)	XPW	Y	N
Semaphore control operations (semsys)	XSE	Y	N

Shared memory control operations (shmsys)	XSH	Y	N
PRIVILEGE			
Grant	PST	N	A
Revoke	PRT	N	A
Create privilege set	PSC	Y	S
Delete privilege set	PSD	Y	S
Add privilege to privilege set	PSA	N	A
Remove privilege from privilege set	PSR	N	A
PROGRAM (see note)			
Load/execute	XLD	Y	Y
Unload	XUL	Y	Y
SAT (see note)			
Command HOLD-SAT-LOGGING	ZHO	N	A
Command RESUME-SAT-LOGGING	ZRE	N	A
Command MODIFY-SAT-PRESELECTION	ZPS	N	A
Command MODIFY-SAT-SUPPORT-PARAMETERS	ZMS	N	A
Command CHANGE-SAT-FILE	ZCH	N	A
Command SAVE-SAT-PARAMETERS	ZSP	N	A
Open SATLOG file (HEADER record)	ZBG	N	A
Close SATLOG file (TRAILER record)	ZND	N	A
SAT event preselection	ZEP	N	A
SAT-ALARM			
Command ADD-SAT-ALARM-CONDITIONS	ZCA	N	A
Command REMOVE-SAT-ALARM-CONDITIONS	ZDA	N	A
Command MODIFY-SAT-ALARM-CONDITIONS	ZMA	N	A
Trigger SAT alarm	ZAL	N	A
SAT-FILTER			
Command ADD-SAT-FILTER-CONDITIONS	ZCF	N	A
Command REMOVE-SAT-FILTER-CONDITIONS	ZDF	N	A

Command MODIFY-SAT-FILTER-CONDITIONS	ZMF	N	A
SESAM (see note)			
Administer DBH session	SEA	Y	N
Change access rights and user accesses	SEP	Y	N
DDL, SSL, utility statement	SES	Y	N
Start/stop SESAM task (DBH or service task)	SET	Y	N
Stop process	SEU	Y	N
SMS (System Managed Storage)			
Create storage class	SCC		
Modify characteristics of storage class	SCM		
Delete storage class	SCD		
Bind storage class to volume set list	SCB		
PVSREN: delete all storage classes	SCP		
Unbind storage class from volume set list	SCU		
Command CHANGE-STORAGE-CLASS-CATALOG	SCX		
Create volume set list	VLC		
Modify volume set list	VLM		
Delete volume set list	VLD	Y	N
Add volume to volume set list	VLA	Y	N
Remove volume from volume set list	VLR	Y	N
Command CHANGE-VOLUME-SET-LIST-CATALOG	VLX	Y	N
PVSREN: rename volume set	VP1	Y	N
PVSREN: rename all volume sets	VP2	Y	N
PVSREN: delete all volume sets	VP3	Y	N
SPOOL DEVICE			
Define RSO device	SDA	Y	N
Modify attributes	SDM	Y	N
Delete entry	SDR	Y	N
SPOOL JOBS (see note)			

Request printing	JPR	Y	N
Delete job	JPC	Y	N
Terminate printing	JPE	Y	N
Interrupt printing	JPI	Y	N
SUBSYSTEM (see note)			
Activate	SCR	Y	A
Deactivate	SDL	Y	A
Hold	SHD	Y	A
Remove	SRM	Y	A
Resume	SRS	Y	A
Connection with nonprivileged subsystem	SCN	Y	N
Disconnection from nonprivileged subsystem	SDS	Y	N
Catalog management	SCT	Y	A
Load subsystem part	SLP	Y	N
Change subsystem file	SFC	Y	N
SYNTAX FILE			
Activate	YAC	Y	N
Modify	YMD	Y	N
Open hierarchy (OPNCALL macro)	YON	Y	N
Activate for subsystem	YAD	Y	N
Check	YCK	Y	N
TAPE encryption			
CREATE-ENCRYPTION-KEY statement	TKC	Y	A
ADD-ENCRYPTION-KEY statement	TKA	Y	A
COPY-ENCRYPTION-KEYS statement	TKP	Y	A
REMOVE-ENCRYPTION-KEYS statement	TKR	Y	A
SHOW-ENCRYPTION-KEYS statement	TKS	Y	N
SET-WRITE-ENCRYPTION-KEY statement	TWK	Y	A
DELETE-KEY-BOX statement	TBD	Y	A

EXPORT-KEY-BOX statement	TBE	Y	A
IMPORT-KEY-BOX statement	TBI	Y	A
REPAIR-KEY-BOX statement	TBR	Y	N
MODIFY-VOLUME-ENCRYPTION-ATTR statement	TVM	Y	A
SHOW-VOLUME-ENCRYPTION-ATTR statement	TVS	Y	N
Access to key box	TBA	Y	A
TERMINAL SET			
Generate	TSB	Y	N
Copy	TSC	Y	N
Delete	TSD	Y	N
Modify	TSM	Y	N
USERID (see note)			
Add	UAD	Y	A
Modify attributes	UMD	Y	N
Remove	URM	Y	A
Lock	ULK	Y	N
Unlock	UUL	Y	S
Check	UCK	Y	F
Define protection attributes	USL	Y	A
Modify protection attributes	UML	Y	A
Modify password protection	UMP	Y	A
Command REQUEST-OPERATOR-ROLE	UOP	Y	A
Command MODIFY-POSIX-USER-ATTRIBUTES	UPA	Y	N
Command MODIFY-POSIX-USER-DEFAULTS	UPD	Y	N
Command MODIFY-USER-PUBSET-ATTRIBUTES	UUP	Y	A
Command MODIFY-LOGON-DEFAULTS	UDM	Y	A
Command SET-LOGON-DEFAULTS	UDS	Y	A
Command UNLOCK-USER-SUSPEND	UUS	Y	A
UTM events (see note)	TRM	Y	A

VOLUME (MAREN) (see note)			
Administrator is modifying attributes	VMA	Y	N
Remove volume	VRM	Y	N
Add volume	VAD	Y	N
User is modifying attributes	VMU	Y	N
User is processing volume	VVP	Y	N
Modify MAREN parameters	VMM	Y	N
Show volume attributes	VSA	Y	N
Show MAREN parameters	VSP	Y	N
VOLUME (other products) (see note)			
Open volume	VON	Y	N
Close volume	VCL	Y	N
Initialize protected volume	VIP	Y	A
Initialize unprotected volume	VIN	Y	N
Initialize disk	VID	Y	A
Install IOCF	VIO	Y	N
Request volume (FDDRL)	VDA	Y	S
Release volume (FDDRL)	VDR	Y	N
Modify volume (FDDRL)	VDU	Y	S

Table 4: Object-related events, event names and audit attributes

Notes on objects and events relating to them

Note on ANY events

The $SATANY macro may be issued by the security administrator and the SAT file manager (using system exit 110) to write to the SATLOG file any information they wish to record about an event that is to be logged (see section “Refining selection with system exit no.110”).

Note on CONSLOG

CLG are not auditable events.
For the purpose of SATUT evaluation, however, it is also possible to use standard format CONSLOG as input files. The entries in these files are converted into CLG records for evaluation, and as a result can be incorporated in the selection. The contents of the audit record are dependent on the type of CONSLOG message (see "Tables of auditable information on object-related events (1)").

Note on DATA SPACES

Operations in the privileged state (TPR) are not logged. If SCOPE=LOCAL is used, failure of the command is logged.

Note on EVENTING-ITEM

If SCOPE = LOCAL applies, no auditing takes place.

Note on FILE and FITC

1. If the audit attribute of a file is activated, all attempted or successful accesses to the file are logged, provided the event result matches the value of the audit attribute (seesection “Subject, object and event”).
2. The following file attributes are security-relevant: user-access, access, audit, passwords, retention period, basic access control list. Since system administration under TSOS is authorized to read the passwords entered in the directory entry, this event (’read password’) is also rated as security-relevant.
3. The following two events may be logged when deleting a file:
   - delete data
   - delete protection attributes

   The same events may occur in conjunction with the renaming of a file with simultaneous modification of the protection attributes.
4. In the event of a single programmed instruction closing all files, the event (’close file’) is recorded separately for each file.
5. In multiprocessor systems, auditing is performed by the computer from which the file was opened, while the shareability and the access rights are checked on the computer on which the file is cataloged.
6. Since ARCHIVE subtasks make use of the Subject Identification Interface (SID), all events relating to a FILE object are treated as if they were part of the main task and therefore logged [5]. In addition, the ARCHIVE-specific event ’rename file’ is logged.
7. The FSO event is used to log user requests, i.e. requests from the job to SPACEOPT.

   The FME event is used to log the result of job processing.

   An FSO event record can be associated with no, one or multiple FME result records depending on whether any, and if so how many, files have been moved during job processing.

   In contrast, a record containing the FME result is always preceded by a record with the FSO result.

Notes on JOB

1. Jobs that are canceled while in the wait state are only logged via the CANCEL command.
2. Print jobs are logged in conjunction with the appropriate commands (see SPOOL JOBS).
3. Job classes are irrelevant for SAT logging.
4. In multiprocessor systems, REMOTE ENTER and REMOTE CANCEL are logged in the target computer, while OPEN is recorded in the source computer.
5. SAT does not log the event ’terminate job’ (JED) unless expressly requested to do so; this is because the event is already logged by CONSLOG and accounting.

Notes on MEMORY-POOL

1. When in privileged mode (TPR), the only logged event is ’Make readable in TU’.

2. For memory pools with SCOPE = LOCAL, the only logged events are 'Modify read access' and 'Make readable in TU'.

Note on POSIX-...

Logging of events for POSIX-CHILD-Process, POSIX-FILE-and-Directory, POSIX-PROCESS and POSIX-System-Resources takes place only if SAT support has been activated for these events:

```
/MODIFY-SAT-SUPPORT-PARAMETERS POSIX-EVENTS=*ENABLED
```

Note on PROGRAM

No SAT logging is performed in the event of SLICE OVERLOADING.

Note on SAT events

SAT events are always logged, and auditing cannot be deactivated for these events even via a selection function. SATLOG files always have a header and a trailer record corresponding to the special events 'start of SATLOG file' (ZBG) and 'end of SATLOG file' (ZND). ZBG and ZND events are likewise always logged and cannot be excluded by means of deactivation.

A header record corresponding to the event "Create an analysis file" is created in the analysis files generated by SATUT (ZRR for replacement files, ZRA for analysis files).

Each event which is related to the definition of alarms or filters (ZCA, ZDA, ZMA, ZAL, ZCF, ZDF, ZMF) is also logged. This also includes saving the SAT parameter file (ZSP).

Note on SESAM

SESAM/SQL Server provides the SESAM administrator with options for switching the SAT logging on and off for SESAM. This means that SESAM events can only occur if
SESAM/SQL Server is being used, and only then if the SAT logging is enabled for SESAM. The settings of the SAT preselection have no effect on the SESAM results in all other cases.

Therefore, to log the SESAM events, both the SAT logging in SESAM and the SESAM events in the SAT preselection must be enabled.

Further information on the SESAM options can be found in the "SESAM/SQL Server Database Operation" manual [33].

Note on SPOOL JOBS

Only the fact that /PRINT and /CANCEL commands have been issued is recorded, i.e. the command execution itself is not recorded.

Notes on SUBSYSTEM

1. Any subsystem activation that takes place prior to SYSTEM READY is not logged in the SATLOG file, but in the CONSLOG. Thus security is ensured by setting system parameter SECSTART=Y (see the "Commands" manual), what forces creation of the CONSLOG file.
2. When a subsystem is accessed, the access request is recorded but not the subsystem operations (since they are performed under a different TSN).
3. Only connection or disconnection requests to nonprivileged subsystems are logged, provided they have been successful.

Notes on USERID

1. The audit data does not indicate whether the authorization to activate AUDIT mode (event UAD or UMD) has been modified.
2. The rejection of interactive and batch jobs is recorded only indirectly by the ’check user ID’ event, since it does not involve any other security-relevant events.

Notes on UTM events

Since the subject of a UTM event usually is not a BS2000 user ID, such events are treated differently by SAT.

SAT only recognizes that a UTM event has occurred. The audit data contains a subcode indicating the specific UTM events.

For detailed information on SAT logging under openUTM, refer to the openUTM manual “Generating Applications” [17].

Notes on VOLUME

1. SAT does not record whether or not the write-enable ring of a magnetic tape was present.
2. SAT records any reservation of a magnetic tape via DMS (see the “Introductory Guide to DMS” [6]), FDDRL (see the “FDDRL” manual [9]) or INIT (see the “Utility Routines” manual [14]). The VSN of a tape being initialized is unknown.
3. SHOW-VOLUME-ATTRIBUTES statements are not considered to be security-relevant events unless they are part of a user job under TSOS or under a user ID possessing the TAPE-ADMINISTRATION privilege.

   MAREN parameters are those parameters to be modified by means of the MAREN statement MODIFY-MAREN-PARAMETERS.

2.8 Tables of auditable information on object-related events (1)

The tables in the following show a list of the events for each object, with the associated information fields and their type of output:

- M = mandatory (is always output)
- O = optional (may be output)
- E = *EXTENDED field (is only output if LOGGING-QUANTITY=*EXTENDED was set with the /MODIFY-SAT-PRESELECTION command)
- - = is not output

The other table accompanying each of these table shows which value may appear in each field.

The field names which can also be monitored via the alarm function of SAT or for which a filter condition can be defined are identified in the second column (al/fil) by means of an asterisk (*) or a plus sign (+). Fields marked with a plus (+) can only be checked for their existence (VALUE=*ALL). The asterisk (*) mark means that the contents of the field can also be checked. If the data type for SAT-ALARM and SAT-FILTER differs from the data type for SATUT, the data type for SAT-ALARM and SAT-FILTER is specified in parentheses ().

The identifier in the third column can be used to edit the SAT information in the audit records when using exit routine 110. The identifiers are specified in hexadecimal notation.

The following fields of the objects are always assigned values in every SATLOG record:

- user ID and TSN of the subject (user-id, tsn)
- logging time (timestp)
- abbreviated name of the event (evt) and result of the event (res)
  The event is always shown in the tables, and the result (S = success, F = failure) only if its contents determine what is logged in the variable part of the SATLOG record.

Depending on its existence, but independently of the object, the following fields are always assigned values:

- The auditid field. This contains:
  - the personal user ID if access by means of personal identification was defined for a user ID.

  If there is no personal user ID:
  - the first eight characters of the Kerberos principal, if the authentication at $DIALOG was performed via Kerberos
- group ID (groupid)

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
auditid	*	0001	Audit subject identification type: x-string 2..32
evt		00F3	Event-type id type: c-string 1..3
groupid	*	0002	Group subject identification (user-group) type: c-string 1..8

res		00F5	Event result keywords: F/S
timestp		00F1	Time (date and time of the record creation) Format: yyyy-mm-dd/hh:mm:ss
tsn		00F4	TSN subject type: c-string 1..4
user-id		00F6	user subject identification type: c-string 1..8 (user-id)

These fields are therefore not mentioned again in the object-specific tables.

Notes

Under certain circumstances there may not be a relevant cause task with events of the BCAM object. The contents of the user-id and tsn fields are meaningless in this case. The string ’(BCAM)’ is entered in the user-id field.

There is no relevant cause task with the IPSEC event “Security policy infringement during data transfer“ (IPV). The contents of the user-id and tsn fields are meaningless in this case. The string ’(IPSEC)’ is entered in the user-id field.

In the case of fields relatingto the SAT objects POSIX-FILE-and-Directory, POSIX-CHILD-Process, POSIX-PROCESS and POSIX-SYSTEM-Resources, question marks (i.e. ‘??’) may be displayed as the field value in the SATLOG log if the input by the user is incorrect.

If the user-id field contains the string ’(OPR)’, the cause of the event is a system task of operating.

For the fields eltvers (version of a library element) and ldvers (version of a loaded module) it is important to note that the largest possible version is not – like, for example, by LMS – converted into the character @, and instead retains its internal coding X'FF'. Therefore, for a query in SATUT, irrespective of the specified character set, the
character ß or ~ has to be used.

Object ADAM (only up to BS2000 OSD/XC V11.0)

Event	evt	SAT information	
		device	devtype
Add device management	ADO	M	M

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
device	*	003C	Device name type: c-string 1..8
devtype	*	003D	Device type type: x-string 2..4 (ALARM/FILTER: x-string 4..4)

Object ANY

Event	evt	SAT information				
		databth	datahex	datatxt	ldata	subcod
ANY event (system exit) Any event	ANY	O	O	O	E	O

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
databth	*	0062	data type hexa or string type: x-string 2..510
datahex	*	0061	data type hexa type: x-string 2..510
datatxt	*	0060	data type text type: c-string 1..510
ldata	*	0203	General hexa fields type: x-string 2..64000
subcod	*	005F	subcode type: c-string 1..4

Object APPLICATION

Event	evt	res	SAT information							
			applnam	hostnam	partnam	pthtnam	parttyp	applid	connid	rc
Open application	DON	S	M	O	-	-	-	M	-	M
		F	M	O	-	-	-	O	-	M
Close application	DCL	S/F	O	-	-	-	-	O	-	O
Connect application	DCN	S	M	O	M	O	M	M	M	M
		F	M	O	M	O	M	O	O	M
Abort application	DDS	S/F	M	O	M	O	-	O	O	O

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
applid	*	0035	Application identifier type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
applnam	*	0025	Application name type: c-string 1..8
connid	*	0036	Connection identifier type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
hostnam	*	0029	Name of the host type: c-string 1..8
partnam	*	0026	Partner name type: c-string 1..8
parttyp	*	0028	Type of the partner keywords: APPLICATION/TERMINAL
pthtnam	*	0027	Name of the partner host type: c-string 1..8
rc	*	003B	fdbk return code type: x-string 2..8 (ALARM/FILTER: x-string 8..8)

Object BCAM

Event	evt	res	SAT information																	
			applid	**appliso**[1]	**applnam**[1]	**applsoc**[1]	**connid**	**hostnam**	**ipv4own**	**ipv4ptn**	**ipv6own**	**ipv6ptn**	**itslown**	**itslptn**	**partiso**[2]	**partnam**[2]	**partsoc**[2]	**portown**	**portptn**	**pthtnam**	**rc**
Open TSAP	BAO	S	M	O	O	O	-	O	-	-	-	-	O	-	-	-	-	O	-	-	M
		F	M	O	O	O	-	O	-	-	-	-	O	-	-	-	-	O	-	-	M
Close TSAP	BAC	S	M	O	O	O	-	O	-	-	-	-	O	-	-	-	-	O	-	-	M
		F	M	-	-	-	-	-	-	-	-	-	O	-	-	-	-	O	-	-	M
Open connection	BCN	S	M	-	-	-	M	-	O	O	O	O	O	O	O	O	O	O	O	O	M
		F	M	-	-	-	-	-	O	O	O	O	O	O	O	O	O	O	O	O	M
Close connection	BDS	S	M	-	-	-	M	-	-	-	-	-	O	-	-	-	-	-	-	-	M
		F	M	-	-	-	M	-	-	-	-	-	O	-	-	-	-	-	-	-	M

[1]The appliso, applnam and applsoc fields are mutually exclusive

[2]The partiso, partnam and partsoc fields are mutually exclusive

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
applid	*	0035	Application identifier type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
appliso	*	0146	ISO name of the application type: x-string 156..156 (ALARM/FILTER: x-string 2..156)
applnam	*	0025	Application name type: c-string 1..8
applsoc	*	0147	SOCKET name of the application type: x-string 156..156 (ALARM/FILTER: x-string 2..156)
connid	*	0036	Connection identifier type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
hostnam	*	0029	Name of the host type: c-string 1..8

ipv4own	*	014A	Own IP address (format V4) type: c-string 7..15
ipv4ptn	*	014B	Partner IP address (format V4) type: c-string 7..15
ipv6own	*	014C	Own IP address (format V6) type: c-string 39..39
ipv6ptn	*	014D	Partner IP address (format V6) type: c-string 39..39
itslown	*	0151	Own ISO-TSEL type: x-string 64..64 (ALARM/FILTER: x-string 2..64)
itslptn	*	0152	Partner ISO-TSEL type: x-string 64..64 (ALARM/FILTER: x-string 2..64)
partiso	*	0148	ISO name of the partner type: x-string 156..156 (ALARM/FILTER: x-string 2..156)
partnam	*	0026	Partner name type: c-string 1..8
partsoc	*	0149	SOCKET name of the partner type: x-string 156..156 (ALARM/FILTER: x-string 2..156)
portown	*	014E	Own port number type: integer 0..65535
portptn	*	014F	Partner port number type: integer 0..65535
pthtnam	*	0027	Name of the partner host type: c-string 1..8
rc	*	003B	fdbk return code type: x-string 2..8 (ALARM/FILTER: x-string 8..8)

Object CATALOG

Event	evt	res	SAT information				
			catid	share	catacce	resjoin	newcat
Import catalog	CIP	S/F	M	M	M	M	-
Export catalog	CEP	S/F	M	-	-	-	-
Check catalog	CKR	S	M	-	-	-	O
		F	O	-	-	-	O
Convert catalog	CVR	S	M	-	-	-	M
		F	O	-	-	-	O

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
catacce	*	0024	Catalog access status keywords: MASTER/SLAVE
catid	*	0022	Catalog identifier type: c-string 1..4
newcat	*	00EC	new or merged catalog identifier type: c-string 1..4
resjoin	*	0045	user-catalog lost or not keywords: YES/NO
share	*	0023	Catalog shareability keywords: SHARED/EXCLUSIVE

Object CONSLOG

Event	evt	SAT information					
		cltype	clrecpt	clsende	clmsgid	cltext	clorig
CONSLOG entry	CLG[1]	CLG[1]	M	-	M	M	M
	CLG[2]	CLG[2]	M	M	-	M	M
	CLG[3]	CLG[3]	-	M	-	M	M

[1]'System message requiring a response' to 'Additional information request', see below
[2]'Response message' and 'Additional information response'
[3]'Operator command'

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
clmsgid		0096	Message id. in Conslog record type: c-string 1..7
clorig		0098	origin of the Conslog record keywords: TPR/TU
clrecpt		0094	Recipient type: c-string 1..4
clsende		0095	Sender type: c-string 1..4
cltext		0097	Rest of the CONSLOG record type: c-string 1..252
cltype		0093	Type of the CONSLOG record type: c-string 1..39. Possible values : 'System message requiring a response' (msg type = ?) 'System message not requiring a response' (msg type = %) 'Error message' (msg type = *) 'Emergency message' (msg type = E) 'Command end message' (msg type = !) 'Command result' (msg type = +) 'Additional information request' (msg type = &) 'Response message' (response type = R) 'Additional information response' (response type = :) 'Operator command' (command type : /)

Object COOWNER PROTECTION

Event	evt	SAT information					
		guard	nwrlnam	objnam	parcra	parcrm	rulenam
Add co-owner protection rule	CRA	M	-	-	E	-	M
Modify co-owner protection rule	CRM	M	O	-	-	E	M
Display co-owner authorization rule	CRQ	-	-	M	-	-	-
Remove co-owner protection rule	CRR	M	-	-	-	-	M
Display co-owner protection rule	CRS	M	-	-	-	-	-

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
guard	*	009F	Guard name type: c-string 1..40
nwrlnam	*	00C1	New name of protection rule type: c-string 1..12
objname	*	00C2	Name of object type: c-string 1..54
parcra	+	0215	Parameter list for add coowner prot rule type: x-string 2..384
parcrm	+	0216	Parameter list for modify coowner prot rule type: x-string 2..384
rulenam	*	00C0	Name of protection rule type: c-string 1..20

Object DATA SPACES

Event	evt	res	SAT information						
			acc key	alet	comread	dsname	mem priv	scope	spid
Create DATA SPACE	DSB	S	M	-	M	M	M	M	M
		F	M	-	M	M	M	M	-
Connect with DATA SPACE	DSC	S	M	M	M	M	M	M	M
		F	M	-	M	M	M	M	M
Release connection with DATA SPACE	DSD	S /F	-	M	-	O	-	O	O
Delete DATA SPACE	DSE	S /F	-	-	-	M	-	M	M
Modify/reset DATA SPACE	DSM	S /F	-	-	-	M	-	M	M

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
acckey	*	0034	TU access key type: x-string 2..2
alet	*	009E	access list entry token type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
comread	*	009D	read access protection type: keywords: NO/YES
dsname	*	009B	data space name type: c-string 1..54
mempriv	*	0021	processed privilege identification keywords: YES/NO
scope	*	001D	Memory pool scope keywords: LOCAL/GROUP/GLOBAL/USER-GROUP/ UNDEFINED/SYSTEM
spid	*	009C	Space identifier type: x-string 2..16 (ALARM/FILTER: x-string 16..16)

Object DEFAULT PROTECTION

Event	evt	SAT information									
		guard	nwrlnam	objnam	pardaa	pardam	pardra	pardrm	pardua	pardur	rulenam
Define default values for protection attributes	DAA	M	-	-	E	-	-	-	-	-	-
Modify default values for protection attributes	DAM	M	-	-	-	E	-	-	-	-	-
Display default values for protection attributes	DAS	M	-	-	-	-	-	-	-	-	-
Add default protection rule	DRA	M	-	-	-	-	E	-	-	-	M
Modify default protection rule	DRM	M	O	-	-	-	-	E	-	-	M
Display default protection attributes for object	DRQ	-	-	M	-	-	-	-	-	-	-
Remove default protection rule	DRR	M	-	-	-	-	-	-	-	-	M
Display default protection rule	DRS	M	-	-	-	-	-	-	-	-	-
Add user ID for object path	DUA	M	-	-	-	-	-	-	E	-	-
Remove user ID for object path	DUR	M	-	-	-	-	-	-	-	E	-
Display user ID for object path	DUS	M	-	-	-	-	-	-	-	-	-

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
guard	*	009F	Guard name type: c-string 1..40
nwrlnam	*	00C1	New name of protection rule type: c-string 1..12
objname	*	00C2	Name of object type: c-string 1..54
pardaa	+	020F	Parameter list for add default prot attributes type: x-string 2..312
pardam	+	0210	Parameter list for modify default prot attributes type: x-string 2..352

pardra	+	0213	Parameter list for add default prot rule type: x-string 2..424
pardrm	+	0214	Parameter list for modify default prot rule type: x-string 2..424
pardua	+	0211	Parameter list for add default prot uid type: x-string 2..952
pardur	+	0212	Parameter list for remove default prot uid type: x-string 2..952
rulenam	*	00C0	Name of protection rule type: c-string 1..20

Object EVENTING-ITEM

Event	evt	res	SAT information	
			evitnam	scope
activate eventing	EEE	S	M	M
		F	-	-
deactivate eventing	EDE	S	M	M
		F	-	-
activate serialization	EES	S	M	M
		F	-	-
deactivate serialization	EDS	S	M	M
		F	-	-

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
evitnam	*	002F	Eventing/serialization item name type: c-string 1..64 (ALARM/FILTER: c-string 64..64)
scope	*	001D	Eventing/serialization item scope keywords: LOCAL/GROUP/GLOBAL/USER-GROUP/ UNDEFINED/SYSTEM

Object FILE

Event	evt	SAT information										
		access	audit at	catid	dmsrc	fil name	fn patrn	fpar cat	new file	pswd par	sop act	vsn1
Create file	FCD	M[1]	M	-	O	M	-	-	-	-	-	-
Read file	FRD	M[2]	M	-	O	M	-	-	-	-	-	-
Execute file (open exec)	FED	M	M	-	O	M	-	-	-	-	-	-
Modify file	FMD	M[3]	M	-	O	M	-	-	-	-	-	-
Close file	FCL	-	M	-	O	M	-	-	-	-	-	-
Delete file	FDD	-	M	-	O	M	-	-	-	-	-	-
Rename file with ARCHIVE	FAR	-	M	-	O	M[5]	-	-	M[5]	-	-	-
Rename file	FRN	-	M	-	O	M	-	-	M	-	-	-
Define protection attributes	FCS	-	M	-	O	M	-	E	-	-	-	-
Modify protection attributes	FMS	-	M	-	O	M	-	E	-	-	-	-
Delete protection attributes	FDS	-	M	-	O	M	-	-	-	-	-	-
Read protection attributes	FRS	-	M	O	-	O[4]	O[4]	-	-	M	-	-
Import protection attributes	FIS	-	M	-	O	M	-	-	-	-	-	-
Export protection attributes	FES	-	M	-	O	M	-	-	-	-	-	-
Convert file into decrypted file	FDC	-	M	-	O	M	-	-	-	-	-	-
Convert file into encrypted file	FEC	-	M	-	O	M	-	-	-	-	-	-
Move file extents (SPACEOPT)	FME	-	-	-	-	M	-	-	-	-	M	O
Select object for reorganization (SPACEOPT)	FSO	-	-	O	-	O	-	-	-	-	-	O

[1]Only OUTIN / OUTPUT permissible

[2]Only INPUT / REVERSE permissible

[3]Only UPDATE / EXTEND / INOUT / SINOUT permissible

[4]Mutually exclusive

[5]Specification of cat-id is contingent upon the ARCHIVE function

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
access	*	0006	Open file mode keywords: INPUT/REVERSE/OUTPUT/EXTEND/UPDATE /INOUT/ OUTIN/SINOUT/INPUT-EXECUTE/UNSPECIFIED
auditat	*	0005	Audit attribute keywords: SUCCESS/FAILURE/ALL/NONE
catid	*	0022	Catalog identifier type: c-string 1..4
dmsrc	*	0004	DMS return code type: x-string 2..4 (ALARM/FILTER: x-string 4..4)
filname	*	0003	File name type: c-string 1..54 (ALARM/FILTER: filename)
fnpatrn	*	0063	filename pattern type: c-string 1..80
fparcat	+	0208	Parameter List File Type: x-string 2..12000
newfile	*	0007	New file name type: c-string 1..54 (ALARM/FILTER: filename)
pswdpar	*	0064	password parameter keywords: YES/NO
sopact	*	0065	SPACEOPT action code keywords: CLEAR-VOL/REDUCE-EXT/ START-JOB
vsn1	*	0039	Volume serial number type: c-string 1..6

Object FITC (Fast Intertask Communication)

Event	evt	SAT information		
		guard	port	tsn-inf
Define port access	POA	M	M	O
Define port	POB	M	M	O
Connect to port	POC	M	M	O
Disconnect from port	POD	M	M	O
Release port	POE	M	M	O
Release port access	POR	M	M	O
Implicit exchange with port	POX	M	M	O

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
guard	*	009F	Guard name type: c-string 1..40
port	*	00B0	port name type: c-string 1..54
tsn-inf	*	0010	TSN object type: c-string 1..4 (ALARM/FILTER: c-string 4..4)

Object GROUP

Event	evt	SAT information					
		admin	catid	gparadu	gparmdu	obj-gid	upper
Add	GAD	M	M	E	-	M	M
Modify	GMD	O	M	-	E	M	O
Remove	GRM	M	M	-	-	M	M
Show	GSH	-	M	-	-	M	-

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
admin	*	002B	(Group)administrator identification type: c-string 1..8 (user-id)
catid	*	0022	Catalog identifier type: c-string 1..4
gparadu	+	0209	Parameter list for add group type: x-string 2..5000
gparmdu	+	0210	Parameter list for modify group type: x-string 2..10000
obj-gid	*	002A	Group identificator as object type: c-string 1..8
upper	*	002C	Upper group identification type: c-string 1..8

Object GUARDS

Event	evt	SAT information				
		catid	gparmod	gparrem	guard	nwguard
Create GUARD	GUB	-	-	-	M	-
Copy GUARD	GUC	-	-	-	M	M
Delete GUARD	GUD	-	-	-	M	-
Change guards catalog	GUF	M	-	-	-	-
Repair guards catalog	GUR	M	-	-	-	-
Modify attributes	GUM	-	-	-	M	O
Display attributes	GUS	-	-	-	M	-
Define access conditions	GAA	-	E	-	M	-
Modify access conditions	GAM	-	E	-	M	-
Delete access conditions	GAR	-	-	E	M	-
Display access conditions	GAS	-	-	-	M	-
Interrogate access conditions	GAQ	-	-	-	M	-

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
catid	*	0022	Catalog identificator type: c-string 1..4
gparmod	+	0201	Modify parameter list type: x-string 2..1860
gparrem	+	0202	delete parameter list type: x-string 2..432
guard	*	009F	Guard name type: c-string 1..40
nwguard	*	00AB	new guard name type: c-string 1..24

Object IPSEC

Event	evt	res	SAT information				
			ipv4own[1]	**ipv4ptn**[2]	**ipv6own** 1	**ipv6ptn** 2	**rc**
Load IPSEC security database Security policy infringement during data transfer	ILD	S	-	-	-	-	-
		F	-	-	-	-	-
	IPV	F[3]	O	O	O	-O	M

[1]The ipv4own and ipv6own fields are mutually exclusive

[2]The ipv4ptn and ipv6ptn fields are mutually exclusive

[3]The result is always F with the IPV event

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
ipv4own	*	014A	Own IP address (format V4) type: c-string 7..15
ipv4ptn	*	014B	Partner IP address (format V4) type: c-string 7..15
ipv6own	*	014C	Own IP address (format V6) type: c-string 39..39
ipv6ptn	*	014D	Partner IP address (format V6) type: c-string 39..39
rc	*	003B	fdbk return code type: x-string 2..8 (ALARM/FILTER: x-string 8..8) The following return codes can be evaluated: X'00400106' Illegal value for SPI (Security Parameter Index) X'00400206' Invalid signature X'00400306' Cryptobox error message: decryption failed X'00400406' Signature/encryption required X'00400506' 1. Security association required for input but not available X'00400606' 2. Security association required for input but not available X'00400706' Invalid security protocol header

Object JOB

Event	evt	res	SAT information												
			calend	endreas	endtype	flush	intime	jobtype	objuid	procnam	rejreas	repeat	rerun	station	tsninf
Initiate batch job or subtask	JBE	S	M	-	-	M	-	M	M	-	-	M	M	-	M
		F	-	-	-	-	-	-	O	-	M	-	-	-	-
Abort job	JCN	S	-	O	-	-	-	-	-	-	-	-	-	-	M
		F	-	O	-	-	-	-	-	-	-	-	-	-	M
Initiate dialog or RLOGIN	JDE	S	-	-	-	-	-	-	-	M	-	-	-	M	-
		F	-	-	-	-	-	-	O	M	M	-	-	M	-
End job	JED	S	-	M	M	-	-	-	-	-	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-	-
Initialize batchjob or subtask	JIN	S	-	-	-	-	M	-	-	-	-	-	-	-	M
		F	-	-	-	-	-	-	-	-	-	-	-	-	-
Modify batch job	JM	S	M	-	-	M	-	-	-	-	-	M	M	-	M
	D	F	-	-	-	-	-	-	-	-	-	-	-	-	-
Generate POSIX task	JFK	S	-	-	-	-	-	-	-	-	-	-	-	-	M
		F	-	-	-	-	-	-	-	-	-	-	-	-	O

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
calend	*	00F0	calendar date for job-start keywords: YES/NO
endreas	*	0017	End reason keywords: ABEND/CANCEL/LOGOFF/MOVE-JOBS/SHUTDOWN
endtype	*	0016	Job termination status keywords: NORMAL/ABNORMAL
flush	*	0014	Job removal keywords: YES/NO

intime	*	001A	Time of the job spoolin type: c-string 1..14 (ALARM/FILTER: c-string 14..14)
jobtype	*	0012	Job type keywords:ENTR = ENTER job, TASK = subtask, MOVE-JOBS = import job description
obj-uid	*	0011	user identificator as object type: c-string 1..8
procnam	*	0019	Processor name type: c-string 1..8 (ALARM/FILTER: c-string 8..8)
rejreas	*	001B	Reject reason keywords: INV-AUTHORIZATION/SYSTEM-ERROR/ CMD-PARAM-ERROR/SATURATION/NO-ERROR
repeat	*	0015	Job start period keywords: YES/NO
rerun	*	0013	Job reinitiation keywords: YES/NO
station	*	0018	Terminal name type: c-string 1..8 (ALARM/FILTER: c-string 8..8)
tsn-inf	*	0010	TSN object type: c-string 1..4 (ALARM/FILTER: c-string 4..4)

Object JOB VARIABLES

Event	evt	SAT information			
		jvname	jvpatrn	newjv	jvsrc
Rename with ARCHIVE	JVA	M	-	M	O
Create prot. attributes	JVC	M	-	-	O
Delete prot. attributes	JVD	M	-	-	O
Modify prot. attributes	JVM	M	-	-	O
Read data	JVG	M	-	-	O
Write data	JVS	M	-	-	O
Query JV	JVQ	O*	O*	-	O
Rename JV	JVR	M	-	M	O

* Mutually exclusive

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
jvname	*	005B	Job variable-name type: c-string 1..54 (ALARM/FILTER: filename)
jvpatrn	*	005E	Job variable-pattern type: c-string 1..80
jvsrc	*	005D	Return-code of job variables x-string 2..4 (ALARM/FILTER: x-string 4..4)
newjv	*	005C	New job variable name type: c-string 1..54 (ALARM/FILTER: filename)

Object KEY

Event	evt	SAT information						
		catid	enctype	kvno	obj-uid	parkrbp	princcl	princsv
Add KERBEROS Encryption Type	KEA	M	M	M	-	-	-	M
Delete KERBEROS Encryption Type	KED	M	M	M	-	-	-	M
Add KERBEROS Principal	KPA	M	-	-	-	E	-	M
Delete KERBEROS Principal	KPD	M	-	-	-	-	-	M
Modify KERBEROS Principal	KPM	M	-	-	-	E	-	M
KERBEROS Ticket Check	KTC[1]	-	M	M	O	-	M	M
Abortive attempt at a crypto password check after exceeding the maximum number of abortive attempts	KXM[2]	-	-	-	-	-	-	-

[1]With the KTC event the data field user-id has no meaning.
With dialog logon the field is not assigned values. In the case of access via an application such as OMNIS or UTM it contains the caller’s user ID, but not the destination user ID.

[2]With the KXM event the result 'S' cannot occur; the result is thus always 'F'.

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
catid	*	0022	Catalog identificator type: c-string 1..4
enctype	*	0174	KERBEROS encryption type type: integer 0..2^{31}-1
kvno	*	0175	KERBEROS key version number type: integer 0..2^{31}-1
obj-uid	*	0011	User identificator as object type: c-string 1..8
parkrbp	*	0219	Parameter list for add/modify KERBEROS principal type: x-string 2..280
princcl	*	0172	KERBEROS client principal type: c-string 1..1800 with-low (ALARM/FILTER: c-string 1..255 with-low)

princsv	*	0173	KERBEROS server principal type: c-string 1..1800 with-low (ALARM/FILTER: c-string 1..255 with-low)

Object MEMORY-POOL

Event	evt	res	SAT information						
			mempool	scope	memclas	mempriv	enamprc	shortid	acckey
Open (ENAMP)	MEN	S	M	M	M	M	M	M	M
		F	M	O	O	O	M	-	O
Close (DISMP)	MDS	S /F	M	M	-	-	-	-	-
Release (RELMP)	MRL	S /F	M	M	-	-	-	-	-
Make readable for TU (with CSTMP in TPR)	MRD	S /F	M	M	-	-	-	-	-
Modify readability with CSTMP in TU	MAC	S /F	M	M	-	-	-	-	-

Field name	al/fil	exit	Meaning and values of information:SDF data type or keywords
acckey	*	0034	TU access key type: x-string 2..2
enamprc	*	001E	ENAMP return code type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
memclas	*	0020	memory class of the memory pool keywords: CLASS5/CLASS6
mempool	*	001C	memory pool name type: c-string 1..54
mempriv	*	0021	Privilege of the mem. pool pages keywords: YES/NO
scope	*	001D	Memory pool scope keywords: LOCAL/GROUP/GLOBAL/USER-GROUP/ UNDEFINED/SYSTEM
shortid	*	001F	Short memory pool name type: x-string 2..8 (ALARM/FILTER: x-string 8..8)

Object OPERATOR-ROLES

Event	evt	SAT information			
		catid	obj-uid	oprole	routcod
Add routing code	ORA	M	-	M	M
Create operator role	ORB	M	-	M	-
Assign operator role	ORC	M	M	M	-
Remove operator role (from user record)	ORD	M	M	M	-
Delete operator role	ORE	M	-	M	-
Withdraw routing code	ORR	M	-	M	M

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
catid	*	0022	Catalog identificator type: c-string 1..4
obj-uid	*	0011	User identificator as object type: c-string 1..8
oprole	*	00AE	operator role type: c-string 1..8
routcod	*	00AD	routing code type: c-string 1..3

Object PLAM members (elements)

Event	evt	res	SAT information									
			fil name	elt name	elt vers	elt type	nwel nam	nwel ver	nwel typ	plam rc	keep opt	auditat
Create library element	LCE	S	M	M	M	M	O	O	O	-	-	M
		F	M	O	O	O	O	O	O	M	-	M
Modify library element	LM	S	M	M	M	M	-	-	-	-	-	M
	E	F	M	O	O	O	-	-	-	M	-	M
Read library element	LRE	S	M	M	M	M	-	-	-	-	-	M
		F	M	O	O	O	-	-	-	M	-	M
Execute library element	LEE	S	M	M	M	M	-	-	-	-	-	M
		F	M	O	O	O	-	-	-	M	-	M
Close library element	LCL	S	M	M	M	M	-	-	-	-	M	M
		F	M	O	O	O	-	-	-	M	M	M
Delete library element	LDE	S	M	M	M	M	-	-	-	-	-	M
		F	M	O	O	O	-	-	-	M	-	M
Rename library element	LRN	S	M	M	M	M	M	M	M	-	-	M
		F	M	O	O	O	M	M	M	M	-	M
Create security attributes	LCS	S	M	O	O	O	-	-	-	-	-	M
		F	M	O	O	O	-	-	-	M	-	M
Delete security attributes	LDS	S	M	O	O	O	-	-	-	-	-	M
		F	M	O	O	O	-	-	-	M	-	M
Modify security attributes	LM	S	M	O	O	O	-	-	-	-	-	M
	S	F	M	O	O	O	-	-	-	M	-	M

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
auditat	*	0005	Audit attribute keywords: SUCCESS/FAILURE/ALL/NONE
eltname	*	0008	Element name type: c-string 1..64

elttype	*	000A	Element type type: c-string 1..8
eltvers	*	0009	Element version type: c-string 1..24
filname	*	0003	File name type: c-string 1..54 (ALARM/FILTER: filename)
keepopt	*	0047	Keep option keywords: YES/NO
nwelnam	*	0042	New/base element name type: c-string 1..64
nweltyp	*	0044	New/base element type type: c-string 1..8
nwelver	*	0043	New/base element version type: c-string 1..24
plamrc	*	0046	PLAM return code type: c-string 1..13 (ALARM/FILTER: x-string 16..16). Format: zzzz/xxxxxxxx zzzz = primary code (decimal) xxxxxxxx= secondary code (hexadecimal) /HELP PLAzzzz can be used to call up information on the corresponding return code

2.9 Tables of auditable information on object-related events (2)

Object POSIX-CHILD-Process

Event	evt	res	SAT information					
			curpid	currgid	curruid	errno	pid	retval
Create new process (fork)	XFK	S	M	M	M	M	M	M
		F	M	M	M	M	O	M
Create new process on rlogin access (rfork)	XRF	S	M	M	M	M	M	M
		F	M	M	M	M	O	M

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
curpid	*	0100	Current process id of the calling process type: integer 0..2^{31}-1
currgid	*	0102	Current real POSIX group id of the calling process type: integer 0..2^{31}-1
curruid	*	0101	Current real POSIX user id of the calling process type: integer 0..2^{31}-1
errno	*	0103	POSIX error number type: integer -2^{31}..2^{31}-1
pid	*	010F	Process id type: integer 0..2^{31}-1
retval	*	0104	POSIX return value type: integer -2^{31}..2^{31}-1

Object POSIX-FILE-and-Directory

Event	evt	res	SAT information																
			accgrp	accmode	accoth	accusr	atflag	cloexec	currpid	currgid	curruid	dirdes1	dirdes2	errno	fappend	fcreat	fctlcmd	fildes	filname
Change current directory (chdir)	XCD	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
Close file (close)	XCL	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	M	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	M	-
Modify file access rights (chmod)	XCM	S	M	-	M	M	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	M	-	M	M	-	-	M	M	M	-	-	M	-	-	-	-	-
Modify a file’s group or owner (chown)	XCO	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
Create new file (creat)	XCR	S	M	-	M	M	-	-	M	M	M	-	-	M	-	-	-	M	-
		F	M	-	M	M	-	-	M	M	M	-	-	M	-	-	-	O	-
Create directory via descriptor (mkdirat)	XDA	S	M	-	M	M	-	-	M	M	M	M	-	M	-	-	-	-	-
		F	M	-	M	M	-	-	M	M	M	M	-	M	-	-	-	-	-
Duplicate file descriptor (dup)	XDP	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	M	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	M	-
Control operation for files (fcntl)	XFC	S	-	O	-	-	-	O	M	M	M	-	-	M	O	O	M	M	-
		F	-	O	-	-	-	O	M	M	M	-	-	M	O	O	M	M	-
Change current directory via descriptor (fchdir)	XFD	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	M	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	M	-
Change file access rights via descriptor (fchmod)	XFM	S	M	-	M	M	-	-	M	M	M	-	-	M	-	-	-	M	-
		F	M	-	M	M	-	-	M	M	M	-	-	M	-	-	-	M	-
Change a file’s owner or group via descriptor (fchown)	XFO	S	-	-	-	-	-	-	M	M	M			M	-	-	-	M	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	M	-
Create a link to a file via descriptor (linkat)	XLA	S	-	-	-	-	M	-	M	M	M	M	M	M	-	-	-	-	-

		F	-	-	-	-	M	-	M	M	M	M	M	M	-	-	-	-	-
Create link to a file (link)	XLN	S	-	-	-	-	-	-	M	M	M			M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
Change a file or link’s owner or group (lchown)	XLO	S	-	-	-	-	-	-	M	M	M			M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-

Event XCD - XLO (part 1)

Event	evt	res	SAT information																		
			filpos	fnoctty	ftrunc	gid	linknam	mapaddr	maplen	mapprot	mapshar	newfdes	newpath	pathnam	retval	sdevrdo	setsgid	setsuid	symbdev	uid
Changecurrent directory (chdir)	XCD	S	-	-	-	-	-	-	-	-	-	-	-	M	M	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	O	M	-	-	-	-	-
Close file (close)	XCL	S	-	-	-	-	-	-	-	-	-	-	-	O	M	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	O	M	-	-	-	-	-
Modify file access rights (chmod)	XCM	S	-	-	-	-	-	-	-	-	-	-	-	M	M	-	M	M	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	O	M	-	M	M	-	-
Modify a file’s group or owner (chown)	XCO	S	-	-	-	M	-	-	-	-	-	-	-	M	M	-	-	-	-	M
		F	-	-	-	M	-	-	-	-	-	-	-	O	M	-	-	-	-	M
Create new file (creat)	XCR	S	-	-	-	-	-	-	-	-	-	-	-	M	M	-	M	M	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	O	M	-	M	M	-	-
Create directory via descriptor (mkdirat)	XDA	S	-	-	-	-	-	-	-	-	-	-	-	M	M	-	M	M	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	O	M	-	M	M	-	-
Duplicate file descriptor (dup)	XDP	S	-	-	-	-	-	-	-	-	-	M	-	O	M	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	O	-	O	M	-	-	-	-	-
Control operation for files (fcntl)	XFC	S	-	O	O	-	-	-	-	-	-	O	-	O	M	-	-	-	-	-
		F	-	O	O	-	-	-	-	-	-	O	-	O	M	-	-	-	-	-
Change current directory via descriptor (fchdir)	XFD	S	-	-	-	-	-	-	-	-	-	-	-	O	M	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	O	M	-	-	-	-	-
Change file access rights via descriptor (fchmod)	XFM	S	-	-	-	-	-	-	-	-	-	-	-	O	M	-	M	M	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	O	M	-	M	M	-	-
Change a file’s owner or group via descriptor (fchown)	XFO	S	-	-	-	M	-	-	-	-	-	-	-	O	M	-	-	-	-	-
		F	-	-	-	M	-	-	-	-	-	-	-	O	M	-	-	-	-	-
Create a link to a file via descriptor (linkat)	XLA	S	-	-	-	-	M	-	-	-	-	-	-	M	M	-	-	-	-	-
		F	-	-	-	-	O	-	-	-	-	-	-	O	M	-	-	-	-	-
Create link to a file (link)	XLN	S	-	-	-	-	M	-	-	-	-	-	-	M	M	-	-	-	-	-

		F	-	-	-	-	O	-	-	-	-	-	-	O	M	-	-	-	-	-
Change a file or link's owner or group (lchown)	XLO	S	-	-	-	M	-	-	-	-	-	-	-	M	M	-	-	-	-	M
		F	-	-	-	M	-	-	-	-	-	-	-	O	M	-	-	-	-	M

Event XCD - XLO (part 2)

Event	evt	res	SAT information																
			acc grp	accmode	accoth	accusr	atflag	cloexec	curpid	curgid	curuid	dirdes1	dirdes2	errno	fappend	fcreat	fctlcmd	fildes	filname
Change file access rights via descriptor (fchmodat)	XMA	S	M	-	M	M	M	-	M	M	M	M	-	M	-	-	-	-	-
		F	M	-	M	M	M	-	M	M	M	M	-	M	-	-	-	-	-
Create directory (mkdir)	XMD	S	M	-	M	M	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	M	-	M	M	-	-	M	M	M	-	-	M	-	-	-	-	-
Map file in virtual memory (mmap)	XMM	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	M	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	M	-
Set protection attributes for file mapping in virtual memory (mprotect)	XMP	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
Mount file system (mount)	XMT	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	M
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	O
Cancel mapping of file in virtual memory (munmap)	XMU	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
Open file via descriptor (openat)	XOA	S	O	M	O	O	-	-	M	M	M	M	-	M	M	M	-	M	-
		F	O	M	O	O	-	-	M	M	M	M	-	M	M	M	-	O	-
Open file (open)	XOP	S	O	M	O	O	-	-	M	M	M	-	-	M	M	M	-	M	-
		F	O	M	O	O	-	-	M	M	M	-	-	M	M	M	-	O	-
Rename file via descriptor (renameat)	XRA	S	-	-	-	-	-	-	M	M	M	M	M	M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	M	M	M	-	-	-	-	-
Remove directory (rmdir)	XRD	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
Rename file (rename)	XRN	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-

		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
Create symbolic link to a file via descriptor (symlinkat)	XSA	S	-	-	-	-	-	-	M	M	M	M	-	M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	M	-	M	-	-	-	-	-
Create symbolic link to a file (symlink)	XSL	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
Delete file or directory via descriptor (unlinkat)	XUA	S	-	-	-	-	M	-	M	M	M	M	-	M	M	-	-	-	-
		F	-	-	-	-	M	-	M	M	M	M	-	M	M	-	-	-	-
Set file bit mask for a process (umask)	XUM	S	M	-	M	M	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	M	-	M	M	-	-	M	M	M	-	-	M	-	-	-	-	-
Delete file (remove/unlink)	XUN	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
Unmount file system (umount)	XUT	S	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
		F	-	-	-	-	-	-	M	M	M	-	-	M	-	-	-	-	-
Change file group or owner via descriptor (fchownat)	XWA	S	-	-	-	-	M	-	M	M	M	M	-	M	-	-	-	-	-
		F	-	-	-	-	M	-	M	M	M	M	-	M	-	-	-	-	-

Event XMA - XWA (part 1)

Event	**evt**	**res**	**SAT information**																	
			filpos	**fnoctty**	**ftrunc**	**gid**	**linknam**	**mapaddr**	**mapplen**	**mapprot**	**mapshar**	**newfdes**	**newpath**	**pathnam**	**retval**	**sdevrdo**	**setsgid**	**setsuid**	**symbdev**	**uid**
Change file access rights via descriptor (fchmodat)	XMA	S	-	-	-	-	-	-	-	-	-	-	-	M	M	-	M	M	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	O	M	-	M	M	-	-
Create directory (mkdir)	XMD	S	-	-	-	-	-	M	M	M	M	-	-	M	M	-	M	M	-	-
		F	-	-	-	-	-	M	M	M	M	-	-	O	M	-	M	M	-	-
Map file in virtual memory (mmap)	XMM	S	M	-	-	-	-	M	M	M	M	-	-	O	M	-	-	-	-	-
		F	M	-	-	-	-	O	M	M	M	-	-	O	M	-	-	-	-	-
Set protection attributes for file mapping in virtual memory (mprotect)	XMP	S	-	-	-	-	-	M	M	M	-	-	-	-	M	-	-	-	-	-
		F	-	-	-	-	-	M	M	M	-	-	-	-	M	-	-	-	-	-
Mount file system (mount)	XMT	S	-	-	-	-	-	-	-	-	-	-	-	M	M	M	-	-	M	-

		F	-	-	-	-	-	-	-	-	-	-	-	O	M	M	-	-	O	-
Cancel mapping of file in virtual memory (munmap)	XMU	S	-	-	-	-	-	M	M	-	-	-	-	-	M	-	-	-	-	-
		F	-	-	-	-	-	M	M	-	-	-		-	M	-	-	-	-	-
Open file via descriptor (openat)	XOA	S	-	M	M	-	-	-	-	-	-	-	-	M	M	-	O	O	-	-
		F	-	M	M	-	-	-	-	-	-	-	-	O	M	-	O	O	-	-
Open file (open)	XOP	S	-	M	M	-	-	-	-	-	-	-	-	M	M	-	O	O	-	-
		F	-	M	M	-	-	-	-	-	-	-	-	O	M	-	O	O	-	-
Rename file via descriptor (renameat)	XRA	S	-	-	-	-	-	-	-	-	-	-	M	M	M	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	M	O	M	-	-	-	-	-
Remove directory (rmdir)	XRD	S	-	-	-	-	-	-	-	-	-	-	-	M	M	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	O	M	-	-	-	-	-
Rename file (rename)	XRN	S	-	-	-	-	-	-	-	-	-	-	M	M	M	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	M	O	M	-	-	-	-	-
Create symbolic link to a file via descriptor (symlinkat)	XSA	S	-	-	-	-	M	-	-	-	-	-	-	M	M	-	-	-	-	-
		F	-	-	-	-	O	-	-	-	-	-	-	O	M	-	-	-	-	-
Create symbolic link to a file (symlink)	XSL	S	-	-	-	-	M	-	-	-	-	-	-	M	M	-	-	-	-	-
		F	-	-	-	-	O	-	-	-	-	-	-	O	M	-	-	-	-	-
Delete file or directory via descriptor (unlinkat)	XUA	S	-	-	-	-	M	-	-	-	-	-	-	-	M	-	-	-	-	-
		F	-	-	-	-	O	-	-	-	-	-	-	-	M	-	-	-	-	-
Set file bit mask for a process (umask)	XUM	S	-	-	-	-	-	-	-	-	-	-	-	-	M	-	M	M	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-	M	-	M	M	-	-
Delete file (remove/unlink)	XUN	S	-	-	-	-	M	-	-	-	-	-	-	-	M	-	-	-	-	-
		F	-	-	-	-	O	-	-	-	-	-	-	-	M	-	-	-	-	-
Unmount file system (umount)	XUT	S	-	-	-	-	-	-	-	-	-	-	-	M	M	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	O	M	-	-	-	-	-
Change file group or owner via descriptor (fchownat)	XWA	S	-	-	-	M	-	-	-	-	-	-	-	M	M	-	-	-	-	M
		F	-	-	-	M	-	-	-	-	-	-	-	O	M	-	-	-	-	M

Event XMA - XWA (part 2)

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
accgrp	*	0127	Access rights for the group keywords: NONE/R/RW/RWX/RX/W/WX/X

accmode	*	0125	Access mode keywords: READ/READ-AND-WRITE/WRITE/SEARCH
accoth	*	0128	Access rights for other users keywords: NONE/R/RW/RWX/RX/W/WX/X
accusr	*	0126	Access rights for the owner keywords: NONE/R/RW/RWX/RX/W/WX/X
atflag	*	0179	Flag for file operations/events keywords: NONE/ AT-SYMLINK-NOFOLLOW (*in case of event XMA or XWA*) / AT-SYMLINK-FOLLOW (*in case of event XLA*) / AT-REMOVEDIR (*in case of event XUA*) / AT-FDCWD (*dirdes1 XDA,XLA,XMA,XOA,XRA,XSA,XUA,XWA, dirdes2 XLA, XRA*)
cloexec	*	013A	Close file on exec keywords: NO/YES
curpid	*	0100	Current process id of the calling process type: integer 0..2^{31}-1
currgid	*	0102	Current real POSIX group id of the calling process type: integer 0..2^{31}-1
curruid	*	0101	Current real POSIX user id of the calling process type: integer 0..2^{31}-1
dirdes1	*	0177	File descriptor for the first directory type: integer 0..2^{31}-1
dirdes2	*	0178	File descriptor for the second directory type: integer 0..2^{31}-1
errno	*	0103	POSIX error number type: integer -2^{31}..2^{31}-1
fappend	*	0136	Append to the end of file keywords: NO/YES
fcreat	*	0137	Create file keywords: NO/YES
fctlcmd	*	012D	File control operation keywords: DUP-FILDES/SET-FILDES-FLAGS/SET-FILMODE-FLAGS
fildes	*	010C	File descriptor type: integer 0..2^{31}-1
filname	*	0003	File name type: c-string 1..54 (ALARM/FILTER: filename)

filpos	*	011B	Offset in mapped file (in multiple of 512) type: integer 0..2^{31}-1
fnoctty	*	0139	No controlling terminal keywords: NO/YES
ftrunc	*	0138	Truncate file length to 0 keywords: NO/YES
gid	*	0114	POSIX group id type: integer 0..2^{31}-1
linknam	*	0107	Link name type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)
mapaddr	*	011C	Memory address of the mapping type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
maplen	*	011D	Length of the mapped area type: integer 0..2^{31}-1
mapprot	*	0129	Access permission for the mapped pages keywords: NONE/R/RW/RWX/RX/W/WX/X
mapshar	*	012A	Visibility of write accesses to the mapped pages keywords: PRIVATE/SHARED
newfdes	*	010D	New file descriptor type: integer 0..2^{31}-1
newpath	*	0106	New name of file resp. directory type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)
pathnam	*	0105	Name of file resp. directory type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)
retval	*	0104	POSIX return value type: integer -2^{31}..2^{31}-1
sdevrdo	*	013D	Symbolic device is read only keywords: NO/YES
setsgid	*	013C	Set the set-group-id bit keywords: NO/YES
setsuid	*	013B	Set the set-user-id bit keywords: NO/YES

symbdev	*	010A	Symbolic device name (/dev/disk/nnnn) type: c-string 1..14 with-low (ALARM/FILTER: posix-pathname 1..14)
uid	*	0111	POSIX user id type: integer 0..2^{31}-1

Object POSIX-PROCESS

Event	evt	res	SA iInformation										
			curlim	curlim2	curpid	currgid	curruid	egid	errno	euid	gid	maxlim	maxlim2
Set effective group number of a process (setegid)	XEG	S/F	-	-	M	M	M	M	M	-	-	-	-
Set effective user number of a process (seteuid)	XEU	S/F	-	-	M	M	M	-	M	M	-	-	-
Execute file (exec)	XEX	S	-	-	M	M	M	-	M	-	-	-	-
		F	-	-	M	M	M	-	M	-	-	-	-
Set maximum number of group memberships for a process (setgroups)	XGR	S/F	-	-	M	M	M	-	M	-	-	-	-
Send signal to process or process group (kill)	XKL	S/F	-	-	M	M	M	-	M	-	-	-	-
Set process limits (ulimit)	XLM	S/F	M	-	M	M	M	-	M	-	-	-	-
Set the real and effective group number of a process (setregid)	XRG	S/F	-	-	M	M	M	M	M	-	-	-	-
Set the real and effective user number of a process (setreuid)	XRU	S/F	-	-	M	M	M	-	M	M	-	-	-
Set the group number of a process (setgid)	XSG	S/F	-	-	M	M	M	-	M	-	M	-	-
Set the process group number (setpgrp)	XSP	S/F	-	-	M	M	M	-	M	-	-	-	-
Set the limit value for a resource (setrlimit)[1]	XSR1	S/F	O	O	M	M	M	-	M	-	-	O	O
Set the user number of a process (setuid)	XSU	S/F	-	-	M	M	M	-	M	-	-	-	-

Object POSIX-PROCESS (part 1)

Event	evt	res	SA iInformation											
			pathnam	pgid	pid	pidrecv	resourc	retval	rgid	ruid	setpcmd	signal	uid	ulimcmd
Set effective group number of a process (setegid)	XEG	S/F	-	-	-	-	-	M	-	-	-	-	-	-
Set effective user number of a process (seteuid)	XEU	S/F	-	-	-	-	-	M	-	-	-	-	-	-
Execute file (exec)	XEX	S	M	-	-	-	-	M	-	-	-	-	-	-
		F	O	-	-	-	-	M	-	-	-	-	-	-
Set maximum number of group memberships for a process (setgroups)	XGR	S/F	-	-	-	-	-	M	-	-	-	-	-	-
Send signal to process or process group (kill)	XKL	S/F	-	-	-	M	-	M	-	-	-	M	-	-
Set process limits (ulimit)	XLM	S/F	-	-	-	-	-	M	-	-	-	-	-	M
Set the real and effective group number of a process (setregid)	XRG	S/F	-	-	-	-	-	M	M	-	-	-	-	-
Set the real and effective user number of a process (setreuid)	XRU	S/F	-	-	-	-	-	M	-	-	-	-	-	-
Set the group number of a process (setgid)	XSG	S/F	-	-	-	-	-	M	-	-	-	-	-	-
Set the process group number (setpgrp)	XSP	S/F	-	O	O	-	-	M	-	-	M	-	-	-
Set the limit value for a resource (setrlimit)[1]	XSR[1]	S/F	-	-	-	-	-	M	-	-	-	-	-	-
Set the user number of a process (setuid)	XSU	S/F	-	-	-	-	-	M	-	-	-	-	M	-

[1]The curlim and curlim2 fields as well as the maxlim and maxlimt2 fields are mutually exclusive.
If the resource field has the value CPU-TIME or NO-OF-FILES, the curlim and maxlim2 are logged; if resource=FILE-SIZE, the curlim2 and maxlim2 fields are logged.

Object POSIX-PROCESS (part 2)

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
curlim	*	0117	Current limit type: integer 0..2^{31}-1
curlim2	*	0141	Current limit (in multiple of 512) type: integer 0..2^{31}-1
curpid	*	0100	Current process id of the calling process type: integer 0..2^{31}-1
currgid	*	0102	Current real POSIX group id of the calling process type: integer 0..2^{31}-1
curruid	*	0101	Current real POSIX user id of the calling process type: integer 0..2^{31}-1
egid	*	0116	Effective POSIX group id type: integer 0..2^{31}-1
errno	*	0103	POSIX error number type: integer -2^{31}..2^{31}-1
euid	*	0113	Effective POSIX user id type: integer 0..2^{31}-1
gid	*	0114	POSIX group id type: integer 0..2^{31}-1
maxlim	*	0118	Maximum limit type: integer 0..2^{31}-1
maxlim2	*	0142	Maximum limit (in multiple of 512) type: integer 0..2^{31}-1
pathnam	*	0105	Name of file resp. directory type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)
pgid	*	0110	Process group id type: integer 0..2^{31}-1
pid	*	010F	Process id type: integer 0..2^{31}-1
pidrecv	*	010E	Process id of receiving process type: integer -2^{31}..2^{31}-1
resourc	*	012C	Resource keywords: CPU-TIME/FILE-SIZE/NO-OF-FILES

retval	*	0104	POSIX return value type: integer $-2^{31}..2^{31}-1$
rgid	*	0115	Real POSIX group id type: integer $0..2^{31}-1$
ruid	*	0112	Real POSIX user id type: integer $0..2^{31}-1$
setpcmd	*	012E	Suboperation for event XSP (set process group id) keywords: SET-PGID/SET-SID/ SET-SID-AND-PGID
signal	*	012B	Signal sent to a process keywords: ABORT/KILL
uid	*	0111	POSIX user id type: integer $0..2^{31}-1$
ulimcmd	*	012F	Suboperation for event XLM (set process limits) keywords: SET-FILE-LIMIT

Object POSIX-SYSTEM-Resources

Event	evt	res	SAT information															
			accgrp	accoth	account	accusr	curpid	curruid	curuid	dltsec	dltusec	errno	gid	homedir	nsems	obj-uid	procnam	pwcmd
Changesystem time (adjtime)	XAJ	S/F	-	-	-	-	M	M	M	M	M	M	-	-	-	-	-	-
Set user attributes (pwent)	XPW	S/F	-	-	O	-	M	M	M	-	-	M	O	O	-	M	O	M
Semaphore control operation (semsys)	XSE	S	O	O	-	O	M	M	M	-	-	M	O	-	O	-	-	-
		F	O	O	-	O	M	M	M	-	-	M	O	-	O	-	-	-
Shared memory control operations (shmsys)	XSH	S/F	O	O	-	O	M	M	M	-	-	M	O	-	-	-	-	-

Object POSIX-SYSTEM-RESOURCES (part 1)

Event	evt	res	SAT information															
			retval	semact	semcmd	semid	semnum	setsgid	setsuid	shell	shmact	shmaddr	shmcmd	shmid	shmrdo	shmsize	uid	userkey
Change systemtime (adjtime)	XAJ	S/F	M	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Set user attributes (pwent)	XPW	S/F	M	-	-	-	-	-	-	O	-	-	-	-	-	-	O	-
Semaphore control operation (semsys)	XSE	S	M	O	M	M	O	O	O	-	-	-	-	-	-	-	O	O
		F	M	O	M	O	O	O	O	-	-	-	-	-	-	-	O	O
Shared memory control operations (shmsys)	XSH	S/F	M	-	-	-	-	O	O	-	O	O	M	O	O	O	O	O

Object POSIX-SYSTEM-RESOURCES (part 2)

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
accgrp	*	0127	Access rights for the group keywords: NONE/R/RW/RWX/RX/W/WX/X

accoth	*	0128	Access rights for other users keywords: NONE/R/RW/RWX/RX/W/WX/X
account	*	010B	Account number type: c-string 1..8
accusr	*	0126	Access rights for the owner keywords: NONE/R/RW/RWX/RX/W/WX/X
curpid	*	0100	Current process id of the calling process type: integer 0..2^{31}-1
currgid	*	0102	Current real POSIX group id of the calling process type: integer 0..2^{31}-1
curruid	*	0101	Current real POSIX user id of the calling process type: integer 0..2^{31}-1
dltsec	*	0119	Delta seconds for adjusting the system time type: integer 0..2^{31}-1
dltusec	*	011A	Delta microseconds for adjusting the system time type: integer 0..2^{31}-1
errno	*	0103	POSIX error number type: integer -2^{31}..2^{31}-1
gid	*	0114	POSIX group id type: integer 0..2^{31}-1
homedir	*	0108	Home directory of a user type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)
nsems	*	0123	Total number of all semaphores type: integer 0..2^{31}-1
obj-uid	*	0011	User identification as object type: c-string 1..8
procnam	*	0019	Processor name type: c-string 1..8 (ALARM/FILTER: c-string 8..8)
pwcmd	*	0130	Suboperation for event XPW (set user attributes) keywords: FORK-WITH-USER-CHANGE/ MOD-POSIX-USER-ATTR/ POSIX-RLOGIN-ACCESS
retval	*	0104	POSIX return value type: integer -2^{31}..2^{31}-1
semact	*	0133	Action code for semaphore control operation keywords: REMOVE-ID/SET-OPTIONS

semcmd	*	0131	Semaphore control operation keywords: CONTROL/GET
semid	*	0121	Id of the semaphore type: integer 0..2^{31}-1
semnum	*	0122	Number of a specific semaphore type: integer 0..2^{31}-1
setsgid	*	013C	Set the set-group-id bit keywords: NO/YES
setsuid	*	013B	Set the set-user-id bit keywords: NO/YES
shell	*	0109	Shell of a user type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)
shmact	*	0134	Action code for shared memory control operation keywords: REMOVE-ID/SET-OPTIONS
shmaddr	*	011E	Address of the shared memory type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
shmcmd	*	0132	Shared memory control operation keywords: ATTACH/CONTROL/DETACH/GET
shmid	*	0120	Id of the shared memory type: integer 0..2^{31}-1
shmrdo	*	013E	Shared memory is read only keywords: NO/YES
shmsize	*	011F	Size of the shared memory type: integer 0..2^{31}-1
uid	*	0111	POSIX user id type: integer 0..2^{31}-1
userkey	*	0143	User-selected numerical key type : integer -2^{31}..2^{31}-1

Object PRIVILEGE

Event	evt	SAT information			
		priv	obj-uid	catid	privset
Grant	PST	O	M	M	O
Withdraw	PRT	O	M	M	O
Create privilege set	PSC	O	-	M	M
Delete privilege set	PSD	-	-	M	M
Add privilege to privilege set	PSA	M	-	M	M
Remove privilege from privilege set	PSR	M	-	M	M

Field name	al /fil	exit	Meaning and values of information: SDF data type or keywords
catid	*	0022	Catalog identificator type: c-string 1..4
obj-uid	*	0011	User identificator as object type: c-string 1..8
priv	*	002D	Processed privilege identification keywords: see “Table of privileges” in the „SECOS - Security Control System - Access Control“ manual [1].
privset	*	00A3	privilege set name type: c-string 1..8

Object PROGRAM

Event	evt	res	SAT information												
			cxtname	filname	eltname	eltvers	eltype	intname	intvers	intdate	loaduni	memclas	ldvers	mempool	scope
Load phase from PAM file	XLD	S	M	M[1]	-	-	-	M	M	M	M	M	-	O	O
Load phase from library	XLD	S	M	M[2]	M	M	M	M	M	M	M	M	M	O	O
Load phase	XLD	F	O	M	O	O	O	O	O	O	O	M	O	O	O
Load module LLM from lib	XLD	S	M	M[2]	M	M	M	M	M	M	M	M	M	O	O
Load module OM from EAM	XLD	S	M	M[3]	M[4]	-	M	M	-	-	M	M	-	O	O
Load module OM from lib	XLD	S	M	M[2]	M	M	M	M	-	-	M	M	M	O	O
Load module	XLD	F	O	M[2]	O	O	O	O	O	O	O	M	-	O	O
Unload	XUL	S	M	-	-	-	-	O	-	-	O	-	O	O	O
		F	M	-	-	-	-	O	-	-	O	-	O	O	O

LLM = link and load moduleOM = object module

[1]Name of the phase

[2]Name of the library file

[3]The output field contains an asterisk (*)

[4]The output field contains blanks

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
cxtname	*	000F	Context name type: c-string 1..32
eltname	*	0008	Element name type: c-string 1..64
elttype	*	000A	Element type type: c-string 1..8
eltvers	*	0009	Element version type: c-string 1..24

filname	*	0003	File name type: c-string 1..54 (ALARM/FILTER: filename)
intdate	*	000D	Internal date type: c-string 1..8 (ALARM/FILTER: c-string 8..8)
intname	*	000B	Internal name type: c-string 1..41
intvers	*	000C	Internal version type: c-string 1..24
ldvers	*	0099	Load unit version type: c-string 1..24
loaduni	*	000E	Load unit name type: c-string 1..32
memclas	*	0020	memory class of the memory pool keywords: CLASS3/CLASS4/CLASS5/CLASS6
mempool	*	001C	memory pool name type: c-string 1..54
scope	*	001D	Memory pool scope keywords: LOCAL/GROUP/GLOBAL/USER-GROUP/ UNDEFINED/SYSTEM

Object SAT

Event	evt	res	SAT information													
			blksize	confnam	cpuid	device	dodvers	evtaud	filname	logquan	newfile	obj-evt	obj-uid	periodd	periodh	prioallo
Command HOLD-SAT-LOGGING	ZHO	S	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Command RESUME-SAT-LOGGING	ZRE	S	-	-	-	-	-	-	M	-	-	-	-	-	-	-
		F	-	-	-	-	-	-	O	-	-	-	-	-	-	-
Command MODIFY-SAT-PRESELECTION	ZPS	S	-	-	-	-	-	O	-	O	-	O	O	-	-	-
		F	-	-	-	-	-	O	-	O	-	O	O	-	-	-
Command MODIFY-SAT-SUPPORT-PARAMETERS	ZMS	S	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Command CHANGE-SAT-FILE	ZCH	S	M	-	-	O	-	-	M	-	-	-	-	O	O	M
		F	O	-	-	O	-	-	O	-	-	-	-	O	O	O
Command SAVE-SAT-PARAMETERS	ZSP	S	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-	-	-
HEADER record	ZBG		-	M	M	-	M	-	O	-	M	-	-	-	-	-
TRAILER record	ZND		-	-	-	-	M	-	O	-	-	-	-	-	-	-
SAT event preselection when selection file open	ZEP	S	-	-	-	-	-	M	-	-	-	M	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Object SAT (part 1)

Event	evt	res	SAT information													
			prexit	reason	rule	savpar	savval	secallo	supppar	supval	sysid	sysnam	sysvers	uaudeef	useraud	vsn1
	ZHO	S	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Command HOLD-SAT-LOGGING		F	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Command RESUME-SAT-LOGGING	ZRE	S	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Command MODIFY-SAT-PRESELECTION	ZPS	S	O	-	O	-	-	-	-	-	-	-	-	O	O	-
		F	O	-	O	-	-	-	-	-	-	-	-	O	O	-
Command MODIFY-SAT-SUPPORT-PARAMETERS	ZMS	S	-	-	-	-	-	-	O	O	-	-	-	-	-	-
		F	-	-	-	-	-	-	O	O	-	-	-	-	-	-
Command CHANGE-SAT-FILE	ZCH	S	-	-	-	-	-	M	-	-	-	-	-	-	-	O
		F	-	-	-	-	-	O	-	-	-	-	-	-	-	O
Command SAVE-SAT-PARAMETERS	ZSP	S	-	-	-	O	O	-	-	-	-	-	-	-	-	-
		F	-	-	-	O	O	-	-	-	-	-	-	-	-	-
HEADER record	ZBG		-	M	-	-	-	-	-	-	M	M	M	-	-	-
TRAILER record	ZND		-	M	-	-	-	-	-	-	-	-	-	-	-	-
SAT event preselection when selection file open	ZEP	S	-	-	-	-	-	-	-	-	-	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Object SAT (part 2)

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
blksize	*	00CD	block size type: integer, sensible values 1..32767
confnam		008B	configuration name type: c-string 1..21
cpuid		008D	cpu identification type: x-string 2..128
device	*	003C	Device name type: c-string 1..8
dodvers		009A	dod version type: c-string 1..7
evtaud	*	00CE	event auditability keywords: SUCCESS/FAILURE/ALL/NONE
filname	*	0003	File name type: c-string 1..54 (ALARM/FILTER: filename)

logquan	*	00F9	sign for logging quantity type: c-string 1..8
newfile	*	0007	New file name type: c-string 1..54 (ALARM/FILTER: filename)
obj-evt	*	00C9	event as object type: c-string 1..3
obj-uid	*	0011	User identificator as object type: c-string 1..8
periodd	*	00EA	number of days for action repetition type: integer 0..255
periodh	*	00EB	number of hours for action repetition type: integer 0..255
prexit	*	00D0	Exit routine activated keywords: YES/NO
priallo	*	00CB	primary allocation type: integer $-2^{31}..2^{31}-1$
reason	*	008C	reason keywords: RESUME-SAT-LOGGING/ CHANGE-SAT-FILE/DMS-ERROR/ HOLD-SAT-LOGGING/SHUTDOWN/PERIODIC-SWITCHING/STARTUP
rule	*	00CA	rule keywords: FILES-BY-EVENT/ INDEPENDENT/UNCHANGED
savpar	*	00D1	saved parameters keywords: ALARM/FILTER/LOGGING-FILE-ATTRIBUTES/ PRESELECTION/SAT-SUPPORT
savval	*	00D2	saved value keywords: CURRENT/STANDARD
secallo	*	00CC	secondary allocation type: integer $-2^{31}..2^{31}-1$
suppar	*	013F	name of SAT support parameter keywords: POSIX-EVENTS
supval	*	0140	value of SAT support parameter keywords: DISABLED / ENABLED
sysid		0088	system-id type: c-string 1..3

sysnam		0089	system-name type: c-string 1..8
sysvers		008A	system version type: c-string 1..4
useraud	*	00CF	user auditability keywords: YES/NO
vsn1	*	0039	volume serial number type: c-string 1..6

Object SAT-ALARM

Event	evt	res	SAT information								
			alm act	alm lim	alm name	alm rep	alm sel	evt aud	obj-fld	obj-evt	obj-uid
Command ADD-SAT-ALARM-CONDITIONS	ZCA	S	-	-	M	-	-	-	-	-	-
		F	-	-	O			-			-
Command REMOVE-SAT-ALARM-CONDITIONS	ZDA	S	-	-	M	-	-	-	-	-	-
		F	-	-	O			-			-
Command MODIFY-SAT-ALARM-CONDITIONS	ZMA	S	O	O	M	O	O	O	O	O	O
		F	O	O	O	O	O	O	O	O	O
(alarm trigger)[1]	ZAL[1]	-	M	-	M	-	-	M	-	M	-

[1]In addition to the specified fields, all fields of the event that triggered the alarm are also logged.

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
almact	*	00D6	alarm action keywords: OPERATOR-MESSAGE/OPERATOR-PAUSE
almlim	*	00D4	alarm time limit (in seconds) type: integer -1..2^{31}-1
almname	*	00D3	alarm name type: c-string 1..8
almrep	*	00D5	alarm repeat type: integer 0..255
almsel	*	00D7	alarm selection keywords: ON/OFF
evtaud	*	00CE	event auditability keywords: SUCCESS/FAILURE/ALL/NONE
obj-evt	*	00C9	event as object type: c-string 1..3 (ALARM/FILTER: c-string 3..3)
obj-fld	*	00D8	object field type: c-string 1..7

obj-uid	*	0011	User identificator as object type: c-string 1..8

Object SAT-FILTER

Event	evt	res	SAT information		
			ftract	ftrname	ftrsel
Command ADD-SAT-FILTER-CONDITIONS	ZCF		M	M	-
Command REMOVE-SAT-FILTER-CONDITIONS	ZDF		-	M	-
Command MODIFY-SAT-FILTER-CONDITIONS	ZMF		M	M	O

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
ftract	*	00FA	Filter Action Logging on/off keywords: NO-RECORDING/RECORDING
ftrname	*	00FB	Filter Name Type : c-string 1..8
ftrsel	*	00FC	Filter selection keywords: OFF/ON

Object SESAM

Event	evt	res	SAT information													
			applnam	appluid	dbhconf	dbhnam	dbname	dbtable	hostnam	schema	sessubc	sessext	stmtctf	stmtcts	utmsct	utmuser
Administer DBH session	SEA	S	M	O	M	M	-	-	M	-	M	M	-	-	M	M
		F	M	O	M	M	-	-	M	-	M	M	-	-	M	M
Change access rights and user accesses	SEP	S	M	O	M	M	M	-	M	-	M	O	-	-	M	M
		F	M	O	M	M	M	-	M	-	M	O	-	-	M	M
DDL, SSL, utility statement	SES	S	M	O	M	M	M	O	M	O	M	O	-	-	M	M
		F	M	O	M	M	M	O	M	O	M	O	-	-	M	M
Start/stop SESAM task (DBH or service task)	SET	S	-	-	M	M	-	-	-	-	M	M	-	-	-	-
		F	-	-	M	M	-	-	-	-	M	M	-	-	-	-
Stop process	SEU	S	M	O	M	M	-	-	M	-	M	-	M	M	M	M
		F	M	O	M	M	-	-	M	-	M	-	M	M	M	M

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
applnam	*	0025	Application name type: c-string 1..8 In case of a TIAM program the field contains the string 'TSN=<tsn>' where <tsn> stands for the tsn of the application program
appluid	*	0162	Application user id type: c-string 1..8
dbhconf	*	0160	DBH configuration identifier type: c-string 1..1
dbhnam	*	015F	DBH name identifier type: c-string 1..1
dbname	*	0165	Logical database name type: c-string 1..18
dbtable	*	0167	Table name in the catalog type: c-string 1..31

hostnam	*	0029	Name of the host type: c-string 1..8
schema	*	0166	Schema name in the catalog type: c-string 1..31
sessubc	*	015E	Subcode for the SESAM events type: c-string 1..4 The following subcodes can be evaluated, depending on the event concerned: **Event SET** STRT: Start of a SESAM DBH task END: End of a SESAM DBH task **Event SEU** END: End of a session **Event SEA** ADM administration stmt **Event SEP** USR: Add/remove/modify users PRI: Grant/revoke rights **Event SES** DDL: DDL stmt SSL: SSL stmt UTI: Utility stmt
sestext	*	0168	Additional information for the SESAM events type: c-string 1..64
stmtctf	*	0164	Number of unsuccessful statements in the session type: integer $0..2^{31}-1$ A statement is unsuccessful if it is not confirmed with "successful completion", "no data" or "rollback".
stmtcts	*	0163	Number of successful statements in the session type: integer $0..2^{31}-1$ A statement is successful if it is confirmed with "successful completion", "no data" or "rollback".
utmsct	*	0161	UTM session counter type: x-string 2..16 (ALARM/FILTER: x-string 16..16)
utmuser	*	0048	User id in UTM application frame type: c-string 1..8

Additional information on separate data fields

(see also the “SESAM/SQL Server – Database Operation” manual [33])

- The SAT log records for all tasks of a database handler can be identified with the dbhnam and dbhconf fields.
- A specific process can be identified with the hostnam, applnam, utmuser and utmsct fields.
- Some fields are not filled or only filled in a particular way with logging of an administration command via /SEND-MSG:

Field name	**Contents**
hostnam	’SESAM’
applnam	’SEND’
utmuser	’MESSAGE’
appluid	not filled

Object SMS

Event	evt	SAT information				
		catid	newvset	storcla	volset	vslist
Create storage class	SCC	M	-	M	-	-
Modify characteristics of storage class	SCM	M	-	M	-	-
Delete storage class	SCD	M	-	M	-	-
Bind storage class to volume set list	SCB	M	-	M	-	M
PVSREN: delete all storage classes	SCP	M	-	M	-	-
Unbind storage class from volume set list	SCU	M	-	M	-	M
Command CHANGE-STORAGE-CLASS-CATALOG	SCX	M	-	-	-	-
Create volume set list	VLC	M	-	-	-	M
Modify volume set list	VLM	M	-	-	-	M
Delete volume set list	VLD	M	-	-	-	M
Add volume to volume set list	VLA	M	-	-	M	M
Remove volume from volume set list	VLR	M	-	-	M	M
Command CHANGE-VOLUME-SET-LIST-CATALOG	VLX	M	-	-	-	-
PVSREN: rename volume set	VP1	M	M	-	M	M
PVSREN: rename all volume sets	VP2	M	-	-	-	M
PVSREN: delete all volume sets	VP3	M	-	-	-	M

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
catid	*	0022	Catalog identificator type: c-string 1..4
newvset	*	00D9	New volume set name type: c-string 1..4
storcla	*	00ED	storage class type: c-string 1..8
volset	*	00EF	volume set type: c-string 1..4

vslist	*	00EE	volume set list type: c-string 1..8

Object SPOOL DEVICE

Event	evt	SAT information			
		device	admin	station	procnam
Define RSO device	SDA	M	O	O	O
Modify attributes	SDM	M	O	O	O
Delete entry	SDR	M	-	-	-

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
admin	*	002B	(Group)administrator identification type: c-string 1..8 (user-id)
device	*	003C	Device name type: c-string 1..8
procnam	*	0019	Processor name type: c-string 1..8 (ALARM/FILTER: c-string 8..8)
station	*	0018	Terminal name type: c-string 1..8 (ALARM/FILTER: c-string 8..8)

Object SPOOL JOBS

Event	evt	res	SAT information																
			destsup	**device**	**dmsrc**	**eltname**	**elttype**	**eltvers**	**erase**	**filname**[1]	**fnpatrn**[1]	**jobcmd**	**jobcopy**	**joberrt**	**joborig**	**jobpage**	**jobterm**	**plamrc**	**tsn-inf**
Request printing	JPR	S	-	-	-	O	O	O	M	M	-	-	-	-	-	-	-	-	M
		F	-	-	O	O	O	O	M	O	O	-	-	-	-	-	-	-	-
Delete job	JPC	S	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	M
		F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	M
Printing finished	JPE	S	M	M	O	O	O	O	O	M	-	M	O	M	M	O	M	-	M
		F	M	M	O	O	O	O	O	M	-	M	O	M	M	O	M	O	M
Interrupt job	JPI	S	M	M	O	O	O	O	O	M	-	M	O	M	M	O	M	-	M
		F	M	M	O	O	O	O	O	M	-	M	O	M	M	O	M	O	M

[1]mutually exclusive

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
destsup	*	00A4	destination (output support) keywords: LOCAL/REMOTE/TAPE
device	*	003C	Device name type: c-string 1..8
dmsrc	*	0004	DMS return code type: x-string 2..4 (ALARM/FILTER: x-string 4..4)
eltname	*	0008	Element name type: c-string 1..64
elttype	*	000A	Element type type: c-string 1..8
eltvers	*	0009	Element version type: c-string 1..24
erase	*	0041	File deletion status keywords: YES/NO

filname	*	0003	File name type: c-string 1..54 (ALARM/FILTER: filename)
fnpatrn	*	0063	filename pattern type: c-string 1..80
jobcmd	*	00A7	job command origin keywords: PRINT
jobcopy	*	00A8	copies number type: integer 0..255
joberrt	*	00AA	detected errors during spoolout keywords: DEVICERR/DMS/ NONE/PLAM/SYSTEM/USER
joborig	*	00A6	job origin keywords: NORMAL/RTCOPY/RTDIRECT
jobpage	*	00A9	printed pages number type: integer : $0..2^{31}-1$
jobterm	*	00A5	Termination or interruption type keywords: ABORT/CANCEL/KEEP NORMAL/RESPOOL/SUSPEND
plamrc	*	0046	PLAM return code type: c-string 1..13 (ALARM/FILTER: x-string 16..16). Format: zzzz/xxxxxxxx zzzz = primary code (decimal) xxxxxxxx= Secondary code (hexadecimal) Information about the corresponding return code can be obtained with /HELP PLAzzzz
tsn-inf	*	0010	TSN object type: c-string 1..4 (ALARM/FILTER: c-string 4..4)

Object SUBSYSTEM

Event	evt	res	SAT information											
			cat	cxtname	libname	repfile	subsnam	subsver	synfile	inffile	msgfile	obj-uid	sscmd	sshold
Activate	SCR	S/F	-	M	-	-	M	M	-	-	-	-	-	-
Deactivate		S	-	-	-	-	M	M	-	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-
Hold	SHD	S/F	-	-	-	-	M	M	-	-	-	-	-	-
Resume	SRS	S/F	-	-	-	-	M	M	-	-	-	-	-	-
Connection to nonprivileged subsystem	SCN	S	-	-	-	-	M	M	-	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-
Disconnection from nonprivileged subsystem	SDS	S	-	-	-	-	M	M	-	-	-	-	-	-
		F	-	-	-	-	-	-	-	-	-	-	-	-
Catalog management	SCT	S/F	M	-	-	-	-	-	-	-	-	-	-	-
Partially load subsystem file	SLP	S/F	-	O	-	-	M	M	-	-	-	-	-	-
Change subsystem file	SFC	S/F	-	O	O	O	M	M	O	O	O	O	O	O
Remove subsystem	SRM	S/F	-	-	-	-	M	M	-	-	-	-	-	-

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
cat	*	0032	New catalog name type: c-string 1..54 (ALARM/FILTER: filename)
cxtname	*	000F	Context name type: c-string 1..32
inffile	*	00E5	information file type: c-string 1..54 (ALARM/FILTER: filename)
libname	*	00A0	object module library type: c-string 1..54 (ALARM/FILTER: filename)
msgfile	*	00E4	message file type: c-string 1..54 (ALARM/FILTER: filename)

obj-uid	*	0011	user identificator as object type: c-string 1..8
repfile	*	00A1	rep file name type: c-string 1..54 (ALARM/FILTER: filename)
sscmd	*	00E7	DSSM commands permission Keyword: ALLOWED/BY-ADMINISTRATOR/FORBIDDEN
sshold	*	00E6	Indicates if subsystem can be deleted / held Keyword: ALLOWED/FORBIDDEN
subsnam	*	0030	name of the subsystem type: c-string 1..8
subsver	*	0031	version of the subsystem type: c-string 1..7
synfile	*	00A2	syntax file name type: c-string 1..54 (ALARM/FILTER: filename)

Object SYNTAX FILE

Event	evt	res	SAT information		
			filname	syntype	sdfcmd
Activate	YAC	S/F	M	M	-
Modify	YMD	S/F	M	M	-
Open hierarchy (macro OPNCALL)	YON	S/F	M	M	-
Activate for subsystem	YAD	S/F	M	M	-
Check	YCK	S	-	-	-
		F	M	M	M

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
filname	*	0003	File name type: c-string 1..54 (ALARM/FILTER: filename)
sdfcmd	*	00AC	SDF command name type: c-string 1..30
syntype	*	002E	Syntax file type keywords:SYSTEM/GROUP/SUBSYSTEM/USER

Object TAPE encryption

Event	evt	SAT information													
		actokb	**dpthinf**	**dpthost**	**hostnam**	**keybox**	**keyid**	**srchost**	**srctsn**	**srcuser**	**tgthost**	**tgtkbox**	**trfkbox**	**volkind**	**vsn1**
TAPE encryption															
CREATE-ENCRYPTION-KEY statement	TKC	-	-	-	M	M	O	-	-	-	-	-	-	-	-
ADD-ENCRYPTION-KEY statement	TKA	-	-	-	M	M	M	-	-	-	-	-	-	O	-
COPY-ENCRYPTION-KEYS statement	TKP	-	-	-	M	M	M	-	-	-	M	M	-	-	-
REMOVE-ENCRYPTION-KEY statement	TKR	-	-	-	M	M	M	-	-	-	-	-	-	-	-
SHOW-ENCRYPTION-KEYS statement	TKS	-	-	-	M	M	M	-	-	-	-	-	-	-	-
SET-WRITE-ENCRYPTION-KEY statement	TWK	-	-	-	M	M	M	-	-	-	-	-	-	-	-
DELETE-KEY-BOX statement	TBD	-	-	-	M	M	-	-	-	-	-	-	-	-	-
EXPORT-KEY-BOX statement	TBE	-	M	O	M	M	-	-	-	-	-	-	M	-	-
IMPORT-KEY-BOX statement	TBI	-	-	M	-	-	-	-	-	-	M	M	M	-	-
REPAIR-KEY-BOX statement	TBR	-	-	-	M	M	-	-	-	-	-	-	-	-	-
MODIFY-VOLUME-ENCRYPTION-ATTR statement	TVM	-	-	-	-	-	M	-	-	-	-	-	-	O	M
SHOW-VOLUME-ENCRYPTION-ATTR statement	TVS	-	-	-	-	-	-	-	-	-	-	-	-	-	M
Access to key box	TBA	M	-	-	M	M	-	M	M	M	-	-	-	-	-

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords

acctokb	*	0070	access mode to the key box keywords: OPEN-READ / OPEN-WRITE / DELETE / REPAIR
dpthinf	*	006B	Information about the depot host keywords: IMPORT-PROCESS-HOST/OWN
dpthost	*	006D	name of the host where the transfer key box is deposited type: c-string 1..8
hostnam	*	0029	Name of the host type: c-string 1..8
keybox	*	0066	Key box name type: c-string 1..54
keyid	*	0067	Identification string for the encryption key type: c-string 1..18 (ALARM/FILTER: c-string 18..18)
srchost	*	006F	Host from which the key box is accessed type: c-string 1..8
srctsn	*	0071	TSN of the accessing subject type: c-string 1..4
srcuser	*	0072	Userid of the accessing subject type: c-string 1..8
tgthost	*	0068	Name of target host type: c-string 1..8
tgtkbox	*	0069	Name of target key box type: c-string 1..54
trfkbox	*	006A	Name of transfer key box type: c-string 1..54
volkind	*	006E	Kind of volumes for which the key can be used resp. for which the encryption attributes are modified keywords: FROM-FOREIGN-MAREN-DOMAIN
vsn1	*	0039	volume serial number type: c-string 1..6

Object TERMINAL SET

Event	evt	SAT information								
		catid	nwtsnam	nwtsown	nwtsscp	partsb	partsm	tsname	tsowner	tsscope
Generate	TSB	M	-	-	-	E	-	M	M	M
Copy	TSC	M	M	M	M	-	-	M	M	M
Delete	TSD	M	-	-	-	-	-	M	M	M
Modify	TSM	M	-	-	-	-	E	M	M	M

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
catid	*	0022	Catalog identificator type: c-string 1..4
nwtsnam	*	00C6	Name of new terminal set type: c-string 1..8
nwtsown	*	00C8	Owner of new terminal set type: c-string 1..8
nwtsscp	*	00C7	Scope of new terminal set keyword: GROUP / SYSTEM / USER
partsb	+	0217	Parameter list for build terminal set type: x-string 2..200
partsm	+	0218	Parameter list for modify terminal set type: x-string 2..6800
tsname	*	00C3	Terminal set name type: c-string 1..8
tsowner	*	00C5	Terminal set owner type: c-string 1..8
tsscope	*	00C4	Terminal set scope keyword: GROUP / SYSTEM / USER

Object USERID

Event	evt	SAT information														
		catid	chkmode	groupnr	obj-uid	oprole	parlog	parpos	princl	procnam	rejr	routcod	station	upaarmup	upaarus	usernr
Add	UAD	M	-	-	M	-	-	-	-	-	-	-	-	-	E	-
Modify attributes	UMD	M	-	-	M	-	-	-	-	-	-	-	-	-	E	-
Remove	URM	M	-	-	M	-	-	-	-	-	-	-	-	-	-	-
Lock	ULK	M	-	-	M	-	-	-	-	-	-	-	-	-	-	-
Unlock	UUL	M	-	-	M	-	-	-	-	-	-	-	-	-	-	-
Check	UCK	-	M	-	M	-	-	-	O	O	O	-	O	-	-	-
Define protection attributes	USL	M	-	-	M	-	-	-	-	-	-	-	-	-	-	-
Modify protection attributes	UML	M	-	-	M	-	E	-	-	-	-	-	-	-	-	-
Modify password protection	UMP	M	-	-	M	-	E	-	-	-	-	-	-	-	-	-
Command REQUEST-OPERATOR-ROLE	UOP	-	-	-	M	O	-	-	-	-	O	O	-	-	-	-
Command MODIFY-POSIX-USER-ATTRIBUTES	UPA	M	-	O	-	-	-	E	-	-	-	-	-	-	-	O
Command MODIFY-POSIX-USER-DEFAULTS	UPD	M	-	O	-	-	-	E	-	-	-	-	-	-	-	O
Command MODIFY-USER-PUBSET-ATTRIBUTES	UUP	-	-	-	-	-	-	-	-	-	-	-	-	E	-	-
Command MODIFY-LOGON-DEFAULTS	UDM	M	-	-	-	-	E	-	-	-	-	-	-	-	-	-
Command SET-LOGON-DEFAULTS	UDS	M	-	-	-	-	E	-	-	-	-	-	-	-	-	-
Command UNLOCK-USER-SUSPEND	UUS	M	-	-	M	-	-	-	-	-	-	-	-	-	-	-

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
catid	*	0022	Catalog identificator type: c-string 1..4
chkmode	*	0033	Check mode keywords: BATCH/DIALOG/FILE-BATCH-NCHKPASS/ FILE-TFT-CHKPASS/ FILE-TFT-NCHKPASS/NET-DIALOG-ACCESS/ OLD/OPERATOR-CONSOLE/OPERATOR-PROGRAM/ OPERATOR-TERMINAL/POSIX-BATCH/POSIX-REMOTE/ POSIX-RLOGIN/UCON
groupnr	*	00E8	Primary group-id of entry in POSIX group catalog type: integer 0..2^{31}-1
obj-uid	*	0011	User identificator as object type: c-string 1..8
oprole	*	00AE	operator role type: c-string 1..8
parlog	+	020D	/SET-LOGON-PROTECTION, /MODIFY-LOGON-PROTECTION, /SET-LOGON-DEFAULTS & /MODIFY-LOGON-DEFAULTS type: x-string 2..19952
parpos	+	020E	/MODIFY-POSIX-USER-ATTRIBUTES & /MODIFY-POSIX-USER-DEFAULTS type: x-string 2..4800
princcl	*	0175	KERBEROS client principal type: c-string 1..1800 with-low
procnam	*	0019	Processor name type: c-string 1..8 (ALARM/FILTER: c-string 8..8)
rejr	*	005A	Reject reason type: x-string 2..8 (ALARM/FILTER: x-string 8..8) Following return codes can be evaluated ('x' represents any value): X'xx01xxxx' Errored parameter list X'xx20xxxx' System error X'xx40xxxx' LOGON rejected (e.g. incorrect password) X'xx80xxxx' LOGON rejected because of temporary resource bottleneck (e.g. memory saturation)
routcod	*	00AD	routing code type: c-string 1..3
station	*	0018	Terminal name type: c-string 1..8 (ALARM/FILTER: c-string 8..8)

uparmup	+	020C	parameter list for /MODIFY-USER-PUBSET-ATTRIBUTES type: x-string 2..296
uparus	+	020B	parameter list for USERID add and modify type: x-string 2..800
usernr	*	00AF	primary owner identification of POSIX resources type: integer $0..2^{31}-1$

Object UTM

Event TRM Subcode in SAT-Information utmsubc	**SAT information**															
	utmuser	**utmappl**	**lterm**	**pterm**	**muxltrm**	**datnam1**	**datnam2**	**datype**	**utmacty**	**command**	**utmobj1**	**utmobj2**	**utmobj3***	**utmobj4**	**utmobj5***	**utmobj6**
User signon (utmsubc=SIGN)	M	M	M	M	O	-	-	-	-	-	M	-	O	-	-	-
Change password (CHANGE-PW)	M	M	-	-	-	-	-	-	-	-	-	-	O	-	-	-
Data access (DATA ACCESS)	M	M	-	-	-	M	O	M	M	-	-	-	O	-	-	-
Administrator cmd. (ADM-CMD)	M	M	O	O	O	-	-	-	-	M	O	O	O	-	-	-
Change access key (CHG-ACC-KEY)	O	O	-	-	-	-	-	-	-	-	-	-	O	-	-	-
Program unit(PU) start (START-PU)	M	M	M	-	-	-	-	-	-	-	M	O	O	-	-	-
Terminate PU (END-PU)	M	M	M	-	-	-	-	-	-	-	M	O	O	-	-	-
Task connection to UTM application on (TASK ON)	-	M	-	-	-	-	-	-	-	-	M	O	O	-	-	-
Task connection to UTM application off (TASK OFF)	-	M	-	-	-	-	-	-	-	-	M	-	O	-	-	-
Preselection command (SEL-CMD)	M	M	O	O	-	-	-	-	-	M	O	O	O	-	-	-
Change program (CHG-PROG)	-	-	-	-	-	-	-	-	-	-	-	-	-	O	O	O
Dynamic application extension (DYN-EXT)*	-	-	-	-	-	-	-	-	-	-	-	-	-	O	O	O

* after the name: field reserved for future extensions

UTM events (part 1)

Event TRM Subcode in SAT-Information utmsubc	**SAT information**															
	utmobj7*	**utmcall**	**applnam**	**tacnam**	**tacnaid**	**utuser2**	**utmmode***	**utmname**	**utmtype***	**utmvers***	**utmtaid**	**utmstat**	**utmreas**	**utmhex.***	**obj-uid**	**user2**
User signon (utmsubc=SIGN)	-	-	O	-	-	-	-	-	-	-	-	-	-	-	O	-

Change password (CHANGE-PW)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	O
Data access (DATA ACCESS)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Administrator cmd. (ADM-CMD)	-	-	O	O	-	O	-	-	-	-	-	-	-	-	-	-
Change access key (CHG-ACC-KEY)	-	M	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Program unit(PU) start (START-PU)	-	-	-	M	M	-	-	-	-	-	-	-	-	-	-	-
Terminate PU (END-PU)	-	-	-	M	M	-	-	-	-	-	-	-	-	-	-	-
Task connection to UTM application on (TASK ON)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Task connection to UTM application off (TASK OFF)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Preselection command (SEL-CMD)	-	-	O	O	-	O	-	-	-	-	-	-	-	-	-	-
Change program (CHG-PROG)	O	-	-	-	-	-	O	O	O	O	O	O	O	O	-	-
Dynamic application extension (DYN-EXT)*	O	-	-	-	-	-	O	O	O	O	O	O	O	O	-	-

* after the name: field reserved for future extensions

UTM events (part 2)

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
applnam	*	0025	Application name type: c-string 1..8
command	*	0052	command of utm-administrator type: c-string 1..8
datnam1	*	004E	data name for DATA-ACCESS type: c-string 1..8
datnam2	*	004F	data name for DATA-ACCESS type: c-string 1..8
dattype	*	0050	data type for DATA-ACCESS keywords : G/T/U/L
lterm	*	004B	lterm-name type: c-string 1..8
muxltrm	*	004D	mux lterm name type: c-string 1..8
obj-uid	*	0011	user identifier as object type: c-string 1..8

pterm	*	004C	pterm-name type: c-string 1..8
tacnaid	*	0058	identifier keywords: G/C/T/D/P
tacnam	*	0057	tac-name (START-PU,END-PU,ADM-CMD) type: c-string 1..8
user2	*	0059	UTM user, object of CHANGE-PW type: c-string 1..8
utmacty	*	0051	for DATA-ACCESS keywords : WRITE/READ/C/D
utmappl	*	0049	application name type: c-string 1..8
utmcall	*	0056	caller's address CHANGE-ACCESS-K type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
utmhex1	*	00BC	data 1 type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
utmhex2	*	00BD	data 2 type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
utmhex3	*	00BE	data 3 type: x-string 2..16 (ALARM/FILTER: x-string 16..16)
utmhex4	*	00BF	data 4 type: x-string 2..16 (ALARM/FILTER: x-string 16..16)
utmmode	*	00B1	mode type: c-string 1..4 (ALARM/FILTER: c-string 4..4)
utmname	*	00B2	name type: c-string 1..32
utmobj1	*	0053	1st object of the command type: c-string 1..8
utmobj2	*	0054	2nd object of the command type: c-string 1..8
utmobj3	*	0055	3rd object of the command type: c-string 1..8
utmobj4	*	00B8	4th object of the command type: c-string 1..8 (ALARM/FILTER: c-string 8..8)
utmobj5	*	00B9	5th object of the command type: c-string 1..8 (ALARM/FILTER: c-string 8..8)

utmobj6	*	00BA	6th object of the command type: c-string 1..64 (ALARM/FILTER: C-string 64..64)
utmobj7	*	00BB	7th object of the command type: c-string 1..64 (ALARM/FILTER: C-string 64..64)
utmreas	*	00B7	error code type: c-string 1..8 (ALARM/FILTER: c-string 8..8)
utmstat	*	00B6	state type: c-string 1..1
utmsubc	*	004A	subcode for the utm event keywords: CHG-ACC-KEY/ CHANGE-PW/SIGN/DATA-ACCESS/ ADM-CMD/START-PU/END-PU/ TASK-ON/TASK-OFF/SEL-CMD/ DYN-EXT/CHG-PROG
utmtaid	*	00B5	transaction id type: x-string 2..8 (ALARM/FILTER: x-string 8..8)
utmtype	*	00B3	type type: c-string 1..1
utmuser	*	0048	User ID in UTM application frame type: c-string 1..8
utmvers	*	00B4	version type: c-string 1..8 (ALARM/FILTER: c-string 8..8)

Object VOLUME

Event	evt	SAT information											
		volume	newown	vsn1	vsn2	filname	dmsrc	share	device	devtype	inifunc	level	bakfunc
Administrator modifies attributes	VMA	M	M	M	-	-	-	-	-	-	-	-	-
Remove volume	VRM	M	-	M	-	-	-	-	-	-	-	-	-
Add volume	VAD	M	-	M	-	-	-	-	-	-	-	-	-
User modifies attributes	VMU	M	-	M	-	-	-	-	-	-	-	-	-
User processing volume	VVP	M	-	M	-	M	-	-	-	-	-	-	-
Modify MAREN parameters	VMM	-	-	-	-	-	-	-	-	-	-	-	-
Show volume attributes	VSA	M	-	M	-	-	-	-	-	-	-	-	-
Show MAREN parameters	VSP	-	-	-	-	-	-	-	-	-	-	-	-
Open volume	VON	-	-	M	-	-	M	M	-	-	-	-	-
Close volume	VCL	-	-	M	-	-	M	-	-	-	-	-	-
Initialize protected volume	VIP	O	O	M	M	-	-	-	M	M	-	-	-
Initialize unprotected volume	VIN	O	O	M	M	-	-	-	M	M	-	-	-
Initialize disk	VID	-	-	M	M	-	-	-	-	M	M	-	-
Install IOCF	VIO	-	-	-	-	-	-	-	-	-	-	M	-
request volume (FDDRL)	VDA	-	-	M	-	-	-	-	M	M	-	-	M
release volume (FDDRL)	VDR	-	-	M	-	-	-	-	M	M	-	-	M
modify volume (FDDRL)	VDU	-	-	M	M	-	-	-	M	M	-	-	M

Field name	al/fil	exit	Meaning and values of information: SDF data type or keywords
bakfunc	*	0040	Backup function (FDDRL) keywords: DISK-DUMP/DISK-COPY/RLOD
device	*	003C	Device name type: c-string 1..8
devtype	*	003D	Device type type: x-string 2..4 (ALARM/FILTER: x-string 4..4)
dmsrc	*	0004	DMS return code type: x-string 2..4 (ALARM/FILTER: x-string 4..4)
filname	*	0003	File name type: c-string 1..54 (ALARM/FILTER: filename)
inifunc	*	003E	Initialization function (VOLIN) keywords: INIT-DISK/FORMAT-DISK
level	*	003F	Level # of the service processor type: x-string 2..2
newown	*	0038	New volume owner type: c-string 1..8
share	*	0023	Share mode keywords: SHARED/EXCLUSIVE
volume	*	0037	Treated owner or volume owner type: c-string 1..8
vsn1	*	0039	volume serial number type: c-string 1..6
vsn2	*	003A	New volume serial number type: c-string 1..6

2.10 Table of auditable information (field names)

Specific items of information are recorded for each object-related event subject to auditing. These are described in detail in “Tables of auditable information on object-related events (1)”.

The table below lists the names of the fields to which auditable information is written in alphabetical order, accompanied by the possible field values.
The field names are used as keywords to access the information recorded by SAT. They are required in the //ADD-SELECTION-CONDITIONS and //SELECT-RECORDS statements in order to select information for editing.

The field names which can also be monitored via the alarm function of SAT or for which a filter condition can be defined are identified in the second column (al/fil) by means of an asterisk (*) or a plus sign (+). Fields marked with a plus (+) can only be checked for their existence (VALUE=*ALL). The asterisk (*) mark means that the contents of the field can also be checked. If the data type for SAT-ALARM and SAT-FILTER differs from the data type for SATUT, the data type for SAT-ALARM and SAT-FILTER is specified in parentheses ().

The identifier in the third column serves to process the SAT information supplied in the audit records if exit routine 110 is used. The identifiers are given as hexadecimal values.

The rightmost column in the table lists all objects in which the respective field name occurs.

Field name	**al /fil**	**exit**	**Meaning and values of information: SDF data type or keywords**	**Objects**
access	*	0006	Open file mode keywords: INPUT/REVERSE/OUTPUT/EXTEND/ UPDATE/INOUT/OUTIN/SINOUT/ INPUT-EXECUTE/UNSPECIFIED	FILE
accgrp	*	0127	Access rights for the group keywords: NONE/R/RW/RWX/RX/W/WX/X	POSIX-FILE, POSIX-SYSTEM
acckey	*	0034	TU access key type: x-string 2..2	MEMORY POOL, DATA SPACES
accmode	*	0125	Access mode keywords: READ/READ-AND-WRITE/WRITE/SEARCH	POSIX-FILE
accoth	*	0128	Access rights for other users keywords: NONE/R/RW/RWX/RX/W/WX/X	POSIX-FILE, POSIX-SYSTEM
account	*	010B	Account number type: c-string 1..8	POSIX-SYSTEM
acctokb	*	0070	access mode to the key box keywords: OPEN-READ / OPEN-WRITE / DELETE / REPAIR	TAPE

accusr	*	0126	Access rights for the owner keywords: NONE/R/RW/RWX/RX/W/WX/X	POSIX-FILE POSIX-SYSTEM
admin	*	002B	(Group)administrator identification type: c-string 1..8 (user-id)	GROUP, SPOOL DEVICE
alet	*	009E	access list entry token type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	DATA SPACES
almact	*	00D6	alarm action keywords: OPERATOR-MESSAGE/ OPERATOR-PAUSE	SAT-ALARM
almlim	*	00D4	alarm time limit (in seconds) type: integer -1..2^{31}-1	SAT-ALARM
almname	*	00D3	alarm name type: c-string 1..8	SAT-ALARM
almrep	*	00D5	alarm repeat type: integer 0..255	SAT-ALARM
almsel	*	00D7	alarm selection keywords: ON/OFF	SAT-ALARM
applid	*	0035	Application identifier type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	APPLICATION, BCAM
appliso	*	0146	ISO name of the application type: x-string 156..156 (ALARM/FILTER: x-string 2..156)	BCAM
applnam	*	0025	Application name type: c-string 1..8	APPLICATION, BCAM, SESAM, UTM
applsoc	*	0147	SOCKET name of the application type: x-string 156..156 (ALARM/FILTER: x-string 2..156)	BCAM
appluid	*	0162	Application user id type: c-string 1..8	SESAM
atflag	*	0179	Flag for file operations/events keywords: NONE/AT-SYMLINK-NOFOLLOW/ AT-SYMLINK-FOLLOW/AT-REMOVEDIR	POSIX-FILE
auditat	*	0005	Audit attribute keywords: SUCCESS/FAILURE/ALL/NONE	ACL, FILE, PLAM

auditid	*	0001	Audit subject identification type: x-string 2..32	Each SATLOG record if personal logon or KERBEROS is used
bakfunc	*	0040	Backup function (FDDRL) keywords: DISK-DUMP/DISK-COPY/RLOD	VOLUME
blksize	*	00CD	block size type: integer, sensible values 1..32767 (ALARM/FILTER: integer 0..32767)	SAT
calend	*	00F0	calendar date for job start keywords: YES/NO	JOB
cat	*	0032	New catalog name type: c-string 1..54 (ALARM/FILTER: filename)	SUBSYSTEM
catacce	*	0024	Catalog access status keywords: MASTER/SLAVE	CATALOG
catid	*	0022	Catalog identificator type: c-string 1..4	GUARDS, GROUP, CATALOG, FILE, KEY, OPERATOR-ROLE, PRIVILEGE, SMS, TERMINAL SET, USERID
chkmode	*	0033	Check mode keywords: BATCH/DIALOG/ FILE-BATCH-NCHKPASS/FILE-TFT-CHKPASS/ FILE-TFT-NCHKPASS/NET-DIALOG-ACCESS/ OLD/OPERATOR-CONSOLE/ OPERATOR-PROGRAM/ OPERATOR-TERMINAL/POSIX-BATCH/ POSIX-REMOTE/POSIX-RLOGIN/UCON	USERID
clmsgid		0096	Message id. in Conslog record type: c-string 1..7	CONSLOG
cloexec	*	013A	Close file on exec keywords: NO/YES	POSIX-FILE

clorig		0098	origin of the Conslog record keywords: TPR/TU	CONSLOG
clrecpt		0094	Recipient type: c-string 1..4	CONSLOG
clsende		0095	Sender type: c-string 1..4	CONSLOG
cltext		0097	Rest of the CONSLOG record type: c-string 1..252	CONSLOG
cltype		0093	Type of the CONSLOG record type: c-string 1..39. Possible values: 'System message requiring a response' (msg type=?) 'System message not requiring a response' (msg type = %) 'Error message' (msg type = *) 'Emergency message' (msg type = E) 'Command end message' (msg type = !) 'Command result' (msg type = +) 'Additional information request' (msg type = &) 'Response message' (response type = R) 'Additional information response' (response type =:) 'Operator command' (command type: /)	CONSLOG
command	*	0052	command of utm-administrator type: c-string 1..8	UTM
comread	*	009D	read access protection keywords: NO/YES	DATA SPACES
confnam		008B	configuration name type: c-string 1..21	SAT
connid	*	0036	Connection identifier type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	APPLICATION, BCAM
cpuid		008D	cpu identification type: x-string 2..128	SAT
curlim	*	0117	Current limit type: integer $0..2^{31}-1$	POSIX-PROCESS
curlim2	*	0141	Current limit (in multiple of 512) type: integer $0..2^{31}-1$	POSIX-PROCESS

curpid	*	0100	Current process id of the calling process type: integer 0..2^{31}-1	POSIX-CHILD, POSIX-PROCESS, POSIX-FILE, POSIX-SYSTEM
currgid	*	0102	Current real POSIX group id of the calling process type: integer 0..2^{31}-1	POSIX-CHILD, POSIX-PROCESS, POSIX-FILE, POSIX-SYSTEM
curruid	*	0101	Current real POSIX user id of the calling process type: integer 0..2^{31}-1	POSIX-CHILD, POSIX-PROCESS, POSIX-FILE, POSIX-SYSTEM
cxtname	*	000F	Context name type: c-string 1..32	PROGRAM, SUBSYSTEM
databth	*	0062	data type hexa or string type: x-string 2..510	ANY
datahex	*	0061	data type hexa type: x-string 2..510	ANY
datatxt	*	0060	data type text type: c-string 1..255	ANY
datnam1	*	004E	data name for DATA-ACCESS type: c-string 1..8	UTM
datnam2	*	004F	data name for DATA-ACCESS type: c-string 1..8	UTM
dattype	*	0050	data type for DATA-ACCESS keywords: G/T/U/L	UTM
dbhconf	*	0160	DBH configuration identifier type: c-string 1..1	SESAM
dbhnam	*	015F	DBH name identifier type: c-string 1..1	SESAM
dbname	*	0165	Logical database name type: c-string 1..18	SESAM

dbtable	*	0167	Table name in the catalog type: c-string 1..31	SESAM
destsup	*	00A4	destination (output support) keywords: LOCAL/REMOTE/TAPE	SPOOL JOB
device	*	003C	Device name type: c-string 1..8	ADAM, SAT, SPOOL DEVICE, VOLUME
devtype	*	003D	Device type type: x-string 2..4 (ALARM/FILTER: x-string 4..4)	ADAM, VOLUME
dirdes1	*	0177	File descriptor for the first directory type: integer $0..2^{31}-1$	POSIX-FILE
dirdes2	*	0178	File descriptor for the second directory type: integer $0..2^{31}-1$	POSIX-FILE
dltsec	*	0119	Delta seconds for adjusting the system time type: integer $0..2^{31}-1$	POSIX-SYSTEM
dltusec	*	011A	Delta microseconds for adjusting the system time type: integer $0..2^{31}-1$	POSIX-SYSTEM
dmsrc	*	0004	DMS return code type: x-string 2..4 (ALARM/FILTER: x-string 4..4)	FILE, VOLUME, SPOOL JOBS
dodvers		009A	dod version type: c-string 1..7	SAT
dpthinf	*	006B	Information about the depot host keywords: IMPORT-PROCESS-HOST/OWN	TAPE
dpthost	*	006D	name of the host where the transfer key box is deposited type: c-string 1.. 8	TAPE
dsname	*	009B	data space name type: c-string 1..54	DATA SPACES
egid	*	0116	Effective POSIX group id type: integer $0..2^{31}-1$	POSIX-PROCESS
eltname	*	0008	Element name type: c-string 1..64	PLAM, PROGRAM, SPOOL JOBS
elttype	*	000A	Element type type: c-string 1..8	PLAM, PROGRAM, SPOOL JOBS

eltvers	*	0009	Element version type: c-string 1..24	PLAM, PROGRAM, SPOOL JOBS
enamprc	*	001E	ENAMP return code type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	MEMORY POOL
enctype	*	0174	KERBEROS encryption type type: integer $0..2^{31}-1$	KEY
endreas	*	0017	End reason keywords: ABEND/CANCEL/LOGOFF/ MOVE-JOBS/SHUTDOWN	JOB
endtype	*	0016	Job termination status keywords: NORMAL/ABNORMAL	JOB
erase	*	0041	File deletion status keywords: YES/NO	SPOOL JOBS
errno	*	0103	POSIX error number type: integer $-2^{31}..2^{31}-1$	POSIX-CHILD POSIX-PROCESS POSIX-FILE POSIX-SYSTEM
euid	*	0113	Effective POSIX user id type: integer $0..2^{31}-1$	POSIX-PROCESS
evitnam	*	002F	Eventing/serialization item name type: c-string 1..64 (ALARM/FILTER: c-string 64..64)	EVENTING ITEM
evt		00F3	Event-type id type: c-string 1..3	Each SATLOG record
evtaud	*	00CE	event auditability keywords: SUCCESS/FAILURE/ALL/NONE	SAT
fappend	*	0136	Append to the end of file keywords: NO/YES	POSIX-FILE
fcreat	*	0137	Create file keywords: NO/YES	POSIX-FILE
fctlcmd	*	012D	File control operation keywords: DUP-FILDES/SET-FILDES-FLAGS/ SET-FILMODE-FLAGS	POSIX-FILE
fildes	*	010C	File descriptor type: integer $0..2^{31}-1$	POSIX-FILE

filname	*	0003	File name type: c-string 1..54 (ALARM/FILTER: filename)	ACL, FILE, PLAM, POSIX-FILE, PROGRAM, SPOOL JOBS, SAT, SYNTAX FILE, VOLUME
filpos	*	011B	Offset in mapped file (in multiple of 512) type: integer 0..2^{31}-1	POSIX-FILE
flush	*	0014	Job removal keywords: YES/NO	JOB
fnoctty	*	0139	No controlling terminal keywords: NO/YES	POSIX-FILE
fnpatrn	*	0063	filename pattern type: c-string 1..80	FILE, SPOOL JOBS
fparcat	+	0208	Parameter List File Type: x-string 2..12000	FILE
ftract	*	00FA	Filter Action Logging on/off keywords: NO-RECORDING/RECORDING	SAT-FILTER
ftrname	*	00FB	Filter Name Type: c-string 1..8	SAT-FILTER
ftrsel	*	00FC	Filter selection keywords: OFF/ON	SAT-FILTER
ftrunc	*	0138	Truncate file length to 0 keywords: NO/YES	POSIX-FILE
gid	*	0114	POSIX group id type: integer 0..2^{31}-1	POSIX-PROCESS, POSIX-FILE, POSIX-SYSTEM
gparadu	+	0209	Parameter list for add group type: x-string 2..5000	GROUP
gparmdu	+	0210	Parameter list for modify group type: x-string 2..10000	GROUP
gparmod	+	0201	Modify parameter list type: x-string 2..1860	GUARDS
gparrem	+	0202	delete parameter list type: x-string 2..432	GUARDS

groupid	*	0002	Group subject identification (user-group) type: c-string 1..8	Each SATLOG record if SRPM is used
groupnr	*	00E8	Primary group-id of entry in POSIX group catalog type: integer $0..2^{31}-1$	USERID
guard	*	009F	Guard name type: c-string 1..40	FITC, GUARDS, COOWNER PROTECTION, DEFAULT PROTECTION
homedir	*	0108	Home directory of a user type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)	POSIX-SYSTEM
hostnam	*	0029	Name of the host type: c-string 1..8	APPLICATION, BCAM, SESAM, TAPE
inffile	*	00E5	information file type: c-string 1..54 (ALARM/FILTER: filename)	SUBSYSTEM
inifunc	*	003E	Initialization function (VOLIN) keywords: INIT-DISK/FORMAT-DISK	VOLUME
intdate	*	000D	Internal date type: c-string 1..8 (ALARM/FILTER: c-string 8..8)	PROGRAM
intime	*	001A	Time of the job spoolin type: c-string 1..14 (ALARM/FILTER: c-string 14..14)	JOB
intname	*	000B	Internal name type: c-string 1..41	PROGRAM
intvers	*	000C	Internal version type: c-string 1..24	PROGRAM
ipv4own	*	014A	Own IP address (format V4) type: c-string 7..15	BCAM, IPSEC
ipv4ptn	*	014B	Partner IP address (format V4) type: c-string 7..15	BCAM, IPSEC
ipv6own	*	014C	Own IP address (format V6) type: c-string 39..39	BCAM, IPSEC

ipv6ptn	*	014D	Partner IP address (format V6) type: c-string 39..39	BCAM, IPSEC
itslown	*	0151	Own ISO-TSEL type: x-string 64..64 (ALARM/FILTER: x-string 2..64)	BCAM
itslptn	*	0152	Partner ISO-TSEL type: x-string 64..64 (ALARM/FILTER: x-string 2..64)	BCAM
jobcmd	*	00A7	job command origin keywords: PRINT	SPOOL JOB
jobcopy	*	00A8	copies number type: integer 0..255	SPOOL JOB
joberrt	*	00AA	detected errors during spoolout keywords: DEVICERR/DMS/ NONE/PLAM/SYSTEM/USER	SPOOL JOB
joborig	*	00A6	job origin keywords: NORMAL/RTCOPY/RTDIRECT	SPOOL JOB
jobpage	*	00A9	printed pages number type: integer 0..2^{31}-1	SPOOL JOB
jobterm	*	00A5	Termination or interruption type keywords: ABORT/CANCEL/KEEP NORMAL/RESPOOL/SUSPEND	SPOOL JOB
jobtype	*	0012	Job type keywords: ENTR = ENTER job TASK = subtask, MOVE-JOBS = import job description	JOB
jvname	*	005B	Job variable-name type: c-string 1..54 (ALARM/FILTER: filename)	JOB VARIABLES
jvpatrn	*	005E	Job variable-pattern type: c-string 1..80	JOB VARIABLES
jvsrc	*	005D	Return-code of Job Variables x-string 2..4 (ALARM/FILTER: x-string 4..4)	JOB VARIABLES
keepopt	*	0047	Keep option keywords: YES/NO	PLAM
keybox	*	0066	Key box name type: c-string 1..54	TAPE

keyid	*	0067	Identification string for the encryption key type: c-string 1..18 (ALARM/FILTER: c-string 18..18)	TAPE
kvno	*	0175	KERBEROS key version number type: integer 0..2^{31}-1	KEY
ldata	+	0203	General hexa fields type: x-string 2..64000	ANY
ldvers	*	0099	Load unit version type: c-string 1..24	PROGRAM
level	*	003F	Level # of the service processor type: x-string 2..2	VOLUME
libname	*	00A0	object module library type: c-string 1..54 (ALARM/FILTER: filename)	SUBSYSTEM
linknam	*	0107	Link name type: type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)	POSIX-FILE
loaduni	*	000E	Load unit name type: c-string 1..32	PROGRAM
logquan	*	00F9	sign for logging quantity type: c-string 1..8	SAT
lterm	*	004B	lterm-name type: c-string 1..8	UTM
mapaddr	*	011C	Memory address of the mapping type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	POSIX-FILE
maplen	*	011D	Length of the mapped area type: integer 0..2^{31}-1	POSIX-FILE
mapprot	*	0129	Access permission for the mapped pages keywords: NONE/R/RW/RWX/RX/W/WX/X	POSIX-FILE
mapshar	*	012A	Visibility of write accesses to the mapped pages keywords: PRIVATE/SHARED	POSIX-FILE
maxlim	*	0118	Maximum limit type: integer 0..2^{31}-1	POSIX-PROCESS
maxlim2	*	0142	Maximum limit (in multiple of 512) type: integer 0..2^{31}-1	POSIX-PROCESS
memclas	*	0020	memory class of the memory pool keywords: CLASS3/CLASS4/CLASS5/CLASS6	MEMORY POOL, PROGRAM

mempool	*	001C	memory pool name type: c-string 1..54	MEMORY POOL, PROGRAM
mempriv	*	0021	Privilege of the mem. pool pages keywords: YES/NO	DATA SPACES, MEMORY POOL
msgfile	*	00E4	message file type: c-string 1..54 (ALARM/FILTER: filename)	SUBSYSTEM
muxltrm	*	004D	mux lterm name type: c-string 1..8	UTM
newcat	*	00EC	New or merged catalog identification type: c-string 1..4	CATALOG
newfdes	*	010D	New file descriptor type: integer $0..2^{31}-1$	POSIX-FILE
newfile	*	0007	New file name type: c-string 1..54 (ALARM/FILTER: filename)	FILE, SAT
newjv	*	005C	New job variable name type: c-string 1..54 (ALARM/FILTER: filename)	JOB VARIABLES
newown	*	0038	New volume owner type: c-string 1..8	VOLUME
newpath	*	0106	New name of file resp. directory type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)	POSIX-FILE
newvset	*	00D9	New volume set name type: c-string 1..4	SMS
nsems	*	0123	Total number of all semaphores type: integer $0..2^{31}-1$	POSIX-SYSTEM
nwelnam	*	0042	New/base element name type: c-string 1..64	PLAM
nweltyp	*	0044	New/base element type type: c-string 1..8	PLAM
nwelver	*	0043	New/base element version type: c-string 1..24	PLAM
nwguard	*	00AB	new guard name type: c-string 1..24	GUARDS

nwrlnam	*	00C1	New name of protection rule type: c-string 1..12	COOWNER PROTECTION, DEFAULT PROTECTION
nwtsnam	*	00C6	Name of new terminal set type: c-string 1..8	TERMINAL SET
nwtsown	*	00C8	Owner of new terminal set type: c-string 1..8	TERMINAL SET
nwtsscp	*	00C7	Scope of new terminal set keyword: GROUP / SYSTEM / USER	TERMINAL SET
obj-evt	*	00C9	event as object type: c-string 1..3 (ALARM/FILTER: c-string 3..3)	SAT-ALARM
obj-fld	*	00D8	object field type: c-string 1..7	SAT-ALARM
obj-gid	*	002A	Group identificator as object type: c-string 1..8	GROUP
obj-uid	*	0011	User identificator as object type: c-string 1..8	JOB, KEY, PRIVILEGE, OPERATOR ROLE, SAT, SUBSYSTEM, USERID, UTM, POSIX-SYSTEM
objname	*	00C2	Name of object type: c-string 1..54	COOWNER PROTECTION, DEFAULT PROTECTION
oprole	*	00AE	operator role type: c-string 1..8	OPERATOR ROLE, USERID
parcra	+	0215	Parameter list for add coowner prot rule type: x-string 2..384	COOWNER PROTECTION
parcrm	+	0216	Parameter list for modify coowner prot rule type: x-string 2..384	COOWNER PROTECTION
pardaa	+	020F	Parameter list for add default prot attributes type: x-string 2..312	DEFAULT PROTECTION
pardam	+	0210	Parameter list for modify default prot attributes type: x-string 2..352	DEFAULT PROTECTION

pardra	+	0213	Parameter list for add default prot rule type: x-string 2..424	DEFAULT PROTECTION
pardrm	+	0214	Parameter list for modify default prot rule type: x-string 2..424	DEFAULT PROTECTION
pardua	+	0211	Parameter list for add default prot uid type: x-string 2..952	DEFAULT PROTECTION
pardur	+	0212	Parameter list for remove default prot uid type: x-string 2..952	DEFAULT PROTECTION
parkrbp	+	0219	Parameter list for add/modify KERBEROS principal type: x-string 2..280	KEY
parlog	+	020D	/SET-LOGON-PROTECTION, /MODIFY-LOGON-PROTECTION, /SET-LOGON-DEFAULTS & /MODIFY-LOGON-DEFAULTS type: x-string 2..19952	USERID
parpos	+	020E	/MODIFY-POSIX-USER-ATTRIBUTES & /MODIFY-POSIX-USER-DEFAULTS type: x-string 2..4800	USERID
partiso	*	0148	ISO name of the partner type: x-string 156..156 (ALARM/FILTER: x-string 2..156)	BCAM
partnam	*	0026	Partner name type: c-string 1..8	APPLICATION, BCAM
partsb	+	0217	Parameter list for build terminal set type: x-string 2..200	TERMINAL SET
partsm	+	0218	Parameter list for modify terminal set type: x-string 2..6800	TERMINAL SET
partsoc	*	0149	SOCKET name of the partner type: x-string 156..156 (ALARM/FILTER: x-string 2..156)	BCAM
parttyp	*	0028	Type of the partner keywords: APPLICATION/TERMINAL	APPLICATION
pathnam	*	0105	Name of file resp. directory type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)	POSIX-PROCESS, POSIX-FILE
periodd	*	00EA	number of day for action repetition type: integer 0..255	SAT

periodh	*	00EB	number of hours for action repetition type: integer 0..255	SAT
pgid	*	0110	Process group id type: integer 0..2^{31}-1	POSIX-PROCESS
pid	*	010F	Process id type: integer 0..2^{31}-1	POSIX-CHILD, POSIX-PROCESS
pidrecv	*	010E	Process id of receiving process type: integer -2^{31}..2^{31}-1	POSIX-PROCESS
plamrc	*	0046	PLAM return code type: c-string 1..13 (ALARM/FILTER: x-string 16..16) Format: zzzz/xxxxxxxx zzzz = primary code (decimal) xxxxxxxx= secondary code (hexadecimal) /HELP-MSG PLAzzzz can be used to obtain information about the corresponding return code	PLAM, SPOOL DEVICE
port	*	00B0	port name type: c-string 1..54	FITC
portown	*	014E	Own port number type: integer 0..65535	BCAM
portptn	*	014F	Partner port number type: integer 0..65535	BCAM
prexit	*	00D0	Exit routine activated keywords: YES/NO	SAT
priallo	*	00CB	primary allocation type: integer -2^{31}..2^{31}-1	SAT
princcl	*	0172	KERBEROS client principal type: c-string 1..1800 with-low (ALARM/FILTER: c-string 1..255 with-low)	KEY, USERID
princsv	*	0173	KERBEROS server principal type: c-string 1..1800 with-low (ALARM/FILTER: c-string 1..255 with-low)	KEY
priv	*	002D	Processed privilege identification keywords: siehe see “Table of privileges” in the „SECOS - Security Control System - Access Control“ manual [1]	PRIVILEGE
privset	*	00A3	privilege set name type: c-string 1..8	PRIVILEGE

procnam	*	0019	Processor name type: c-string 1..8 (ALARM/FILTER: c-string 8..8)	JOB, USERID, SPOOL DEVICE, POSIX-SYSTEM
pswdpar	*	0064	password parameter keywords: YES/NO	FILE
pterm	*	004C	pterm-name type: c-string 1..8	UTM
pthtnam	*	0027	Name of the partner host type: c-string 1..8	APPLICATION, BCAM
pwcmd	*	0130	Suboperation for event XPW (set user attributes) keywords: FORK-WITH-USER-CHANGE/ MOD-POSIX-USER-ATTR/ POSIX-RLOGIN-ACCESS	POSIX-SYSTEM
rc	*	003B	fdbk return code type: x-string 2..8 (ALARM/FILTER: x-string 8..8) The following return codes can be evaluated with the IPSEC object: • X'00400106' Illegal value for SPI (Security Parameter Index) • X'00400206' Invalid signature • X'00400306' Cryptobox error message: decryption failed • X'00400406' Signature/encryption required • X'00400506' 1. Security association required for input but not available • X'00400606' 2. Security association required for input but not available • X'00400706' Invalid security protocol header	APPLICATION, BCAM, IPSEC
reason	*	008C	reason keywords: RESUME-SAT-LOGGING/CHANGE-SAT-FILE/DMS-ERROR/HOLD-SAT-LOGGING/ PERIODIC-SWITCHING/SHUTDOWN/STARTUP	SAT

rejr	*	005A	Reject reason type: x-string 2..8 (ALARM/FILTER: x-string 8..8) The following return codes can be evaluated ('x' can be any value): X'xx01xxxx' Incorrect parameter list X'xx20xxxx' System error X'xx40xxxx' LOGON rejected (e.g. incorrect password) X'xx80xxxx' LOGON rejected due to temporary resource bottleneck (e.g. memory saturated)	USERID
rejreas	*	001B	Reject reason keywords: INV-AUTHORIZATION/ SYSTEM-ERROR/CMD-PARAM-ERROR/ SATURATION/NO-ERROR	JOB
repeat	*	0015	Job start period keywords: YES/NO	JOB
repfile	*	00A1	rep file name type: c-string 1..54 (ALARM/FILTER: filename)	SUBSYSTEM
rerun	*	0013	Job reinitiation keywords: YES/NO	JOB
res		00F5	Event result keywords: F/S	Each SATLOG record
resjoin	*	0045	user-catalog lost or not keywords: YES/NO	CATALOG
resourc	*	012C	Resource keywords: CPU-TIME/FILE-SIZE/NO-OF-FILES	POSIX-PROCESS
retval	*	0104	POSIX return value type: integer $-2^{31}..2^{31}-1$	POSIX-CHILD, POSIX-PROCESS, POSIX-FILE, POSIX-SYSTEM
rgid	*	0115	Real POSIX group id type: integer $0..2^{31}-1$	POSIX-PROCESS
routcod	*	00AD	routing code type: c-string 1..3	OPERATOR ROLE, USERID
ruid	*	0112	Real POSIX user id type: integer $0..2^{31}-1$	POSIX-PROCESS

rule	*	00CA	rule keywords: FILES-BY-EVENT/ INDEPENDENT/UNCHANGED	SAT
rulenam	*	00C0	Name of protection rule type: c-string 1..20	COOWNER PROTECTION, DEFAULT PROTECTION
savpar	*	00D1	saved parameters keywords: ALARM/FILTER/LOGGING-FILE-ATTRIBUTES /PRESELECTION/ SAT-SUPPORT	SAT
savval	*	00D2	saved value keywords: CURRENT/STANDARD	SAT
schema	*	0166	Schema name in the catalog type: c-string 1..31	SESAM
scope	*	001D	Eventing/serialization item scope Memory pool scope keywords: LOCAL/GROUP/GLOBAL/ USER-GROUP/UNDEFINED/SYSTEM	DATA SPACES, EVENTING ITEM, MEMORY POOL, PROGRAM
sdevrdo	*	013D	Symbolic device is read only keywords: NO/YES	POSIX-FILE
sdfcmd	*	00AC	SDF command name type: c-string 1..30	SYNTAX FILE
secallo	*	00CC	secondary allocation type: integer $-2^{31}..2^{31}-1$	SAT
semact	*	0133	Action code for semaphore control operation keywords: REMOVE-ID/SET-OPTIONS	POSIX-SYSTEM
semcmd	*	0131	Semaphore control operation keywords: CONTROL/GET	POSIX-SYSTEM
semid	*	0121	Id of the semaphore type: integer $0..2^{31}-1$	POSIX-SYSTEM
semnum	*	0122	Number of a specific semaphore type: integer $0..2^{31}-1$	POSIX-SYSTEM

sessubc	*	015E	Subcode for the SESAM events type: c-string 1..4 The following subcodes can be evaluated, depending on the event concerned: **Event SET** • STRT: Start of a SESAM DBH task • END: End of a SESAM DBH task **Event SEU** • END: End of a session **Event SEA** • ADM Administration stmt **Event SEP** • USR: Add/remove/modify users • PRI: Grant/revoke rights **Event SES** • DDL: DDL stmt • SSL: SSL stmt • UTI: Utility stmt	SESAM
sestext	*	0168	Additional information for the SESAM events type: c-string 1..64	SESAM
setpcmd	*	012E	Suboperation for event XSP (set process group id) keywords: SET-PGID/SET-SID/ SET-SID-AND-PGID	POSIX-PROCESS
setsgid	*	013C	Set the set-group-id bit keywords: NO/YES	POSIX-FILE, POSIX-SYSTEM
setsuid	*	013B	Set the set-user-id bit keywords: NO/YES	POSIX-FILE, POSIX-SYSTEM
share	*	0023	Catalog shareability Share mode keywords: SHARED/EXCLUSIVE	CATALOG, VOLUME
shell	*	0109	Shell of a user type: c-string 1..1024 with-low (ALARM/FILTER: posix-pathname 1..255)	POSIX-SYSTEM

shmact	*	0134	Action code for shared memory control operation keywords: REMOVE-ID/SET-OPTIONS	POSIX-SYSTEM
shmaddr	*	011E	Address of the shared memory type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	POSIX-SYSTEM
shmcmd	*	0132	Shared memory control operation keywords: ATTACH/CONTROL/DETACH/GET	POSIX-SYSTEM
shmid	*	0120	Id of the shared memory type: integer $0..2^{31}-1$	POSIX-SYSTEM
shmrdo	*	013E	Shared memory is read only keywords: NO/YES	POSIX-SYSTEM
shmsize	*	011F	Size of the shared memory type: integer $0..2^{31}-1$	POSIX-SYSTEM
shortid	*	001F	Short memory pool name type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	MEMORY POOL
signal	*	012B	Signal sent to a process keywords: ABORT/KILL	POSIX-PROCESS
sopact	*	0065	SPACEOPT action code keywords: CLEAR-VOL/REDUCE-EXT/ START-JOB	FILE
spid	*	009C	Space identifier type: x-string 2..16 (ALARM/FILTER: x-string 16..16)	DATA SPACES
srchost	*	006F	Host from which the key box is accessed type: c-string 1..8	TAPE
srctsn	*	0071	TSN of the accessing subject type: c-string 1..4	TAPE
srcuser	*	0072	Userid of the accessing subject type: c-string 1..8	TAPE
sscmd	*	00E7	DSSM commands permission Keywords: ALLOWED/BY-ADMINISTRATOR/ FORBIDDEN	SUBSYSTEM
sshold	*	00E6	Indicates if subsystem can be deleted / held Keywords: ALLOWED/FORBIDDEN	SUBSYSTEM
station	*	0018	Terminal name type: c-string 1..8 (ALARM/FILTER: c-string 8..8)	JOB, USERID, SPOOL DEVICE

stmtctf	*	0164	Number of unsuccessful statements in the session type: integer 0..2^{31}-1	SESAM
stmtcts	*	0163	Number of successful statements in the session type: integer 0..2^{31}-1	SESAM
storcla	*	00ED	storage class type: c-string 1..8	SMS
subcod	*	005F	subcode type: c-string 1..4	ANY
subsnam	*	0030	Name of the subsystem type: c-string 1..8	SUBSYSTEM
subsver	*	0031	Version of the subsystem type: c-string 1..7	SUBSYSTEM
suppar	*	013F	name of SAT support parameter keywords: POSIX-EVENTS	SAT
supval	*	0140	value of SAT support parameter keywords: DISABLED / ENABLED	SAT
symbdev	*	010A	Symbolic device name (/dev/disk/nnnn) type: c-string 1..14 with-low (ALARM/FILTER: posix-pathname 1..14)	POSIX-FILE
synfile	*	00A2	syntax file name type: c-string 1..54 (ALARM/FILTER: filename)	SYNTAX FILE, SUBSYSTEM
syntype	*	002E	Syntax file type keywords: SYSTEM/GROUP/SUBSYSTEM/USER	SYNTAX FILE
sysid		0088	system-id type: c-string 1..3	SAT, SUBSYSTEM
sysnam		0089	system-name type: c-string 1..8	SAT, SUBSYSTEM
sysvers		008A	system version type: c-string 1..4	SAT, SUBSYSTEM
tacnaid	*	0058	Kennzeichen keywords: G/C/T/D/P	UTM
tacnam	*	0057	tac-name (START-PU,END-PU,ADM-CMD) type: c-string 1..8	UTM
tgthost	*	0068	Name of target host type: c-string 1..8	TAPE
tgtkbox	*	0069	Name of target key box type: c-string 1..54	TAPE

timestp		00F1	Time (date and time of the record creation) Format: yyyy-mm-dd/hh:mm:ss	Each SATLOG record
trfkbox	*	006A	Name of transfer key box type: c-string 1..54	TAPE
tsn		00F4	TSN subject type: c-string 1..4	Each SATLOG record
tsn-inf	*	0010	TSN object type: c-string 1..4 (ALARM/FILTER: c-string 4..4)	JOB, FITC, SPOOL JOBS
tsname	*	00C3	Terminal set name type: c-string 1..8	TERMINAL SET
tsowner	*	00C5	Terminal set owner type: c-string 1..8	TERMINAL SET
tsscope	*	00C4	Terminal set scope keyword: GROUP / SYSTEM / USER	TERMINAL SET
uauddef		00FD	Default of user audit attribute keywords: OFF/ON	SAT
uid	*	0111	POSIX user id type: integer $0..2^{31}-1$	POSIX-PROCESS, POSIX-FILE, POSIX-SYSTEM
ulimcmd	*	012F	Suboperation for event XLM (set process limits) keywords: SET-FILE-LIMIT	POSIX-PROCESS
uparmup	+	020C	parameter list for /MODIFY-USER-PUBSET-ATTRIBUTES type: x-string 2..296	USERID
uparus	+	020B	parameter list for USERID add and modify type: x-string 2..800	USERID
upper	*	002C	Upper group identification type: c-string 1..8	GROUP
user-id		00F6	user subject identification type: c-string 1..8 (user-id)	Each SATLOG record
useraud	*	00CF	user auditability keywords: YES/NO	SAT
userkey	*	0143	User-selected numerical key type : integer $-2^{31}..2^{31}-1$	POSIX-SYSTEM

usernr	*	00AF	primary owner identification of POSIX resources type: integer 0..2^{31}-1	USERID
user2	*	0059	UTM user, object of CHANGE-PW type: c-string 1..8	UTM
utmacty	*	0051	for DATA-ACCESS keywords: WRITE/READ/C/D	UTM
utmappl	*	0049	application name type: c-string 1..8	UTM
utmcall	*	0056	caller's address CHANGE-ACCESS-KEY type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	UTM
utmhex1	*	00BC	data 1 type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	UTM
utmhex2	*	00BD	data 2 type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	UTM
utmhex3	*	00BE	data 3 type: x-string 2..16 (ALARM/FILTER: x-string 16..16)	UTM
utmhex4	*	00BF	data 4 type: x-string 2..16 (ALARM/FILTER: x-string 16..16)	UTM
utmmode	*	00B1	mode type: c-string 1..4 (ALARM/FILTER: c-string 4..4)	UTM
utmname	*	00B2	name type: c-string 1..32	UTM
utmobj1	*	0053	1st object of the command type: c-string 1..8	UTM
utmobj2	*	0054	2nd object of the command type: c-string 1..8	UTM
utmobj3	*	0055	3rd object of the command type: c-string 1..8	UTM
utmobj4	*	00B8	4th object of the command type: c-string 1..8 (ALARM/FILTER: c-string 8..8)	UTM
utmobj5	*	00B9	5th object of the command type: c-string 1..8 (ALARM/FILTER: c-string 8..8)	UTM
utmobj6	*	00BA	6th object of the command type: c-string 1..64 (ALARM/FILTER: c-string 64..64)	UTM

utmobj7	*	00BB	7th object of the command type: c-string 1..64 (ALARM/FILTER: c-string 64..64)	UTM
utmreas	*	00B7	error code type: c-string 1..8 (ALARM/FILTER: c-string 8..8)	UTM
utmsct	*	0161	UTM session counter type: x-string 2..16 (ALARM/FILTER: x-string 16..16)	SESAM
utmstat	*	00B6	state type: c-string 1..1	UTM
utmsubc	*	004A	subcode for the utm event keywords: CHG-ACC-KEY/ CHANGE-PW/ SIGN/DATA-ACCESS/ADM-CMD/START-PU/ END-PU/TASK-ON/TASK-OFF/SEL-CMD/ DYN-EXT/CHG-PROG	UTM
utmtaid	*	00B5	transaction id type: x-string 2..8 (ALARM/FILTER: x-string 8..8)	UTM
utmtype	*	00B3	type type: c-string 1..1	UTM
utmuser	*	0048	User id in UTM application frame type: c-string 1..8	SESAM, UTM
utmvers	*	00B4	version type: c-string 1..8 (ALARM/FILTER: c-string 8..8)	UTM
volkind	*	006E	Kind of volumes for which the key can be used resp. for which the encryption attributes are modified keywords: FROM-FOREIGN-MAREN-DOMAIN	TAPE
volset	*	00EF	volume set type: c-string 1..4	SMS
volume	*	0037	Treated owner or volume owner type: c-string 1..8	VOLUME
vslist	*	00EE	volume set list type: c-string 1..8	SMS
vsn1	*	0039	volume serial number type: c-string 1..6	FILE, SAT, TAPE, VOLUME
vsn2	*	003A	New volume serial number type: c-string 1..6	VOLUME

Table 5: Loggable information: field names, values and objects

3 Glossary

The following glossary contains definitions and explanations of terms that are used within this manual in connection with the description of functional units.

access authorization

Defines the subjects that are permitted to access an object and also the type of access permitted.

access rights

Rights assigned to a subject granting it a defined type of access to an object.

access type

General meaning: the access type defines the way in which an object may be accessed.

The following access types exist for files: read, write and execute access.

The following access types exist for job variables: read and write access.

The access type relating to memory pools is ’enable memory pool’ (ENAMP).The access type relating to serialization is ’enable serialization ID’ (ENASI).

The access type relating to eventing is ’enable eventing ID’ (ENAEI).

account number

Designates an account for a user ID. Any one account number can be assigned to more than one user ID; any one user ID can be assigned more than one (up to 60) account numbers. The account number is evaluated during SET-
LOGON-PARAMETERS (resp. LOGON) and ENTER-JOB.

assurance level

Hierarchical classification with regard to the assurance (quality) of an IT system. In the evaluation, the assurance of an IT system is rated. On the basis of this rating, classification at one of the assurance levels Q0 to Q7 takes place.

attribute guard

Special *guard* in which the default values for object protection attributes are stored.

auditing

Basic function of a secure system, denoting the logging of operations and the editing of the recorded data.

authentication

Evidence of the claimed identity.

authorized user

Subject authorized to access an object, e.g. a user ID authorized to access a file.

BACL

see *basic access control list*

basic access control list (BACL)

Entries in the file directory which determine the access rights for files and job variables (read, write and execute access) assigned to the object owner, the owner's user group and all other user IDs. (Not to be confused with the access control list, ACL.)

catalog ID

Pubset identifier consisting of a maximum of 4 characters <cat-id 1...4>.

command profile

see *profile*

co-owner

User ID that the *owner* of an *object* authorizes to co-administer his/her *object*.

co-ownership

Authorization to co-administer other user's *objects*.

co-owner protection

Special access protection for *objects* that can be co-administered by other user IDs

co-owner protection rule

Rule, applying to one or more *objects*, which defines the conditions a user ID must fulfil in order to be a *co-owner* of these *objects*.

CONSLOG file

Logging file in which the entire message traffic taking place between operator terminals, authorized user programs and the system is recorded.

data access control

Data access control refers to the rules regulating the access of subjects to the objects of a DP system, as well as to the methods used to ensure that these rules are actually observed.

data privacy

In its narrower sense as defined in the Federal Data Protection Act, data privacy denotes the actions and measures necessary to counteract any impairment of the confidential interests of the individual citizen by protecting his or her personal data against the inappropriate use of data processing.

In a broader sense, data privacy denotes the actions and measures necessary to counteract any impairment of one's own confidential interests or those of others by protecting data against inappropriate use at the various stages of data processing.

Within a company or institution, data privacy is put into practice by

- observing the relevant principles and guidelines set up by the company or institution itself
- observing the prevailing legal regulations
- exercising due awareness of the problems involved
- applying data protection measures in accordance with the proclaimed purpose.

data protection

Designates the technical and organizational actions and measures necessary to safeguard the security of data and data processing operations. This involves in particular

- restricting data access to authorized users
- preventing the undesired or unauthorized processing of data
- preventing data corruption during processing
- ensuring data reproducibility.

This task is performed by

- implementing technical and organizational precautions and measures in both hardware and software
- taking other organizational as well as physical and personnel precautions and measures.

default protection

Protection mechanism used to make default settings for protection attributes.

default protection rule

Rule, applying to one or more *objects*, which defines what protection attributes these *objects* have by default.

file directory (catalog)

File that exists on each pubset (in the case of SM pubsets, on each volume set).
Each file and each job variable of a pubset is entered in the appropriate file directory. Files on private disks and tapes may be entered in the file directory.
A directory entry contains all the attributes (protection attributes, location of managed data etc.) of a file or job variable except the access control list.

filter

Mechanism for refining the preselection for SAT.

first start

The first start incorporates the creation of new system files, a number of system user IDs (e.g. TSOS, SYSPRIV, SYSDUMP, SERVICE, SYSGEN, SYSNAC, SYSHSMS, SYSUSER, SYSSNAP, SYSSPOOL, SYSAUDIT) and the JOIN file.
There are two alternative ways of executing a first start for a specific pubset: either system start with this pubset or IMCAT processing (logical addition of a pubset).

function accumulation (combination)

In order to avoid function accumulation, any ADD-USER-GROUP or MODIFY-USER-GROUP command will be rejected that specifies the designation as a group administrator on a particular pubset of a user ID which already possesses the USER-ADMINISTRATION privilege on that pubset or on the home pubset. Similarly, any attempt to assign the USER-ADMINISTRATION privilege to a user ID on a particular pubset (SET-PRIVILEGE) will be rejected if that user ID has already been designated as a group administrator on that pubset.

functionality class

Set of specific minimum requirements as to the functionality of security functions which an IT system is expected to satisfy.
The various functionality classes have been defined in the "Criteria for the Evaluation of Trustworthiness of Information Technology (IT) Systems", 1st Version 1989, published by the German Information Security Agency on behalf of the Government of the Federal Republic of Germany.

global privileges

All the privileges that can be assigned by means of the SET-PRIVILEGE command, as well as the privilege of the security administrator and the
privileges assigned to the TSOS user ID. A detailed list of these privileges can be found under "System administrator privileges".
'Global privileges' and 'system administrator privileges' are synonymous.

global user administration

All those user IDs which are assigned the global privilege USER-ADMINISTRATION.

group administrator

User whose user ID is authorized, via assignment of the group administrator privilege, to manage the group potential, group members and the subordinate group structure. The user ID that is assigned the group administrator privilege is recorded in the group potential of its group.

group administrator privilege

Authorizes a user ID to manage the user IDs of its own group, subordinate user groups, and individual user groups of a hierarchically lower level. Three variants of the group administrator privilege exist, which differ in the scope of activities permitted: MANAGE-RESOURCES, MANAGE-MEMBERS and MANAGE-GROUPS.

group entry

Records in the JOIN file (old name: $TSOS.TSOSJOIN, new name see *user catalog*), containing information on a user group.

group ID

Name of a user group which is assigned when creating the user group. It is used to address the user group.

group member

User ID within a user group. The group administrator can assign individual group members resources from the group potential.

group potential

Contains all the resources and user rights defined for a user group that can be allocated or assigned to the members of that user group or to subordinate user groups.

guard

Protection profile that can be set up and administered using the *GUARDS* protection mechanism.

GUARDS

(Generally Usable Access contRol aDministration System):
Universal protection mechanism for objects in BS2000.

identification

Method of determining the identity of a person or object.

installation

- The process of placing hardware and software in location so that operation is possible.
- The hardware and software set up at a particular user's site.

IT security criteria

see *security criteria*

JOIN file (user catalog)

System file created on each pubset which contains the attributes of the user IDs that are authorized to use the pubset.
If stored on disks initialized with a PAM key, the JOIN file actually consists of two files: $TSOS.TSOSJOIN and $TSOS.SYSSRPM.
If stored on disks initialized without a PAM key, the JOIN file is identical with the file $TSOS.SYSSRPM.

object

Passive element of a DP system which contains or receives information and to which operations such as reading, writing, execution etc. can be applied.

Examples: files, job variables, user IDs, *terminal sets*.

offline mode

- A functional unit is in offline mode if it is not under the direct control of the CPU.
- Operating mode of a device that is neither under the control of nor connected up with a computer (as opposed to online mode).

online mode

- A functional unit is in online mode if it is under the direct control of the CPU.
- Operating mode which permits users to work interactively with a computer.
- Operating mode in which users have access to a computer via data display terminals.
- Operating mode of a device that is either under the control of or connected up with a computer (as opposed to offline mode).

operator role

A set of routing codes collected together under one name. Any desired combination of the 40 routing codes is possible.

owner

User ID under which an *object* is set up.

password

Character string which the user has to enter in order to be granted access under a user ID or access rights for a file, job variable, node or application.
User ID-specific passwords are used for user authentication and thus for system access control, while file-specific passwords are used for verifying access authorizations relating to a file (or job variable) and thus for data access control.

personal audit for individual accountability

Function which ensures the reproducibility of operations in a DP system. Identification mechanism based on any of the following three principles: definition of one user ID per user or restriction of a user's system access to a specific terminal.

personal identification

Other user IDs apart from the current user ID may be authorized to perform access. During the interactive access check, a personal identification/ authentication is performed. The user ID specified with the user-specific
identification is taken over into the SAT entries. In this way, it is possible to trace individual actions to specific users

privilege

Global right which provides authorization for the execution of certain commands and activation of certain program interfaces (e.g. SECURITY-ADMINISTRATION)

privilege set

A set of global privileges which can be addressed with a freely selectable name.

profile

Set of commands which a user ID is authorized to use by means of a syntax file.

protection attributes

Security-relevant attributes of an object which determine the type and scope of access to this object. Files can have the following protection attributes:
ACCESS/USER-ACCESS, SERVICE bit, AUDIT attribute (NONE/SUCCESS/FAILURE/ALL), RDPASS, WRPASS, EXPASS, RETPD, BACL, ACL.

public space

Named disk storage area available to a defined number of user IDs in the operating system. Public space can extend over one or more pubsets.

pubset

Set of public disk storage units defined by a catalog ID.
A distinction is made between single-feature pubsets (SF pubsets) and system-managed pubset (SM pubset).
An SF pubset comprises one or more disks which must be matching in respect of their essential characteristics (disk format, allocation unit, availability).
By contrast, an SM pubset may comprise a number of so-called volume sets having differing characteristics. The essential characteristics of the disks only need to be matching within a volume set.

retention period

Period of time during which the modification or deletion of an object (e.g. a file) is prohibited.

role

Grouping of attributes assigned to a subject, e.g. the role of the security administrator.

rule

Entry in a *rule container.*
A distinction is made between *co-ownership rules* and *default protection rules* depending on their purpose.

rule container

Special guard which contains *co-ownership rules* or *default protection rules*.

SAT
Security Audit Trail

Logging of security-related events.

SATLOG file

SAT log file in which SATCP records security-relevant events.

secure BS2000 system

BS2000 system that is the result of a secure generation.
Synonyms: ’F2/Q3 system’ or ’evaluated system’. The opposite of a ’secure BS2000 system’ is not an ’ insecure BS2000 system’, but rather a system that may include non-evaluated components, that does not satisfy the F2/Q3 criteria, or whose mode of operation does not conform with the recommended configuration.

secure generation

Generation of a BS2000 system that makes active use of all security-relevant parameter settings which guarantee system security.

secure hardware configuration

Installed hardware (including telecommunication devices and network) that is not subject to any security constraints.

security administrator

- In the traditional sense: organizational/administrative institution responsible for security.
- The user ID for the security administrator can be selected with the aid of the startup parameter service. By default, the security administrator has the user ID SYSPRIV. The security administrator is authorized to assign global privileges to user IDs and to withdraw such privileges, as well as to activate/deactivate auditing via SAT, to administer operator roles and to select user IDs and events for auditing.

security criteria

Criteria used to assess the security of information technology (IT) systems.
They comprise functionality classes and assurance levels and are represented as Fx/Qy (functionality class x and assurance level y); F2/Q3, for instance, denotes functionality class 2 and assurance level 3.

session

Operations/activities taking place between system startup and system shutdown.

SF pubset

Single-feature pubset, see *pubset*

single-feature pubset

see *pubset*

Single Sign On

Mechanism which permits access to various computers and applications after a one-off identification /authentication. This access is controlled by certificates.

SM pubset

System-managed pubset, see *pubset*

SMS

System-managed storage; concept for pubset management.

SRPM (System Resources and Privileges Management)

In BS2000, resources and privileges are usually administered from the TSOS user ID. SRPM allows these tasks to be approved for other user IDs as well, in other words it makes it possible to distribute the tasks.

subject

Active element of a DP system that may be the originator of such operations as reading, writing, execution etc., i.e. of operations resulting in an information flow or in a change in the system status (e.g. user ID, program, program section).

system access class

SECOS distinguishes between the following system access classes:

- DIALOG-ACCESS (access in interactive mode)
- NET-DIALOG-ACCESS (interactive access from the network)
- BATCH-ACCESS (access by remote batch terminals)
- OPERATOR-ACCESS-TERM (operating mode)
- OPERATOR-ACCESS-PROG (operating mode for programmed operators)
- OPERATOR-ACCESS-CONS (console access)
- POSIX-RLOGIN-ACCESS (POSIX remote login)
- POSIX-REMOTE-ACCESS (POSIX remote command access)

system access control

This covers all the methods that serve to protect a DP system against unauthorized access.

system administration

- Structural unit of a computer center.
- Persons in control of user IDs that have been assigned global privileges.

system administrator privileges

see *global privileges*

system-managed pubset

see *pubset*

system resources

Resources of a computer system that can be requested/released by a job or task.

system shutdown

Orderly system termination (including backup of special system files).

system startup

Loading of operating system software. The following types of system startup are distinguished:

- dialog startup
- fast startup
- automatic startup

These types of system startup differ in their degree of automation.

terminal

I/O device consisting of a keyboard and a screen and connected to a host computer via network software. The terminal may be connected to the host either directly (via a local cluster controller) or indirectly via a communication computer (in which case it is addressed via a station or transport system address).

terminal set

The purpose of terminal sets is to permit the effective administration of the various terminals via which interactive mode access to a user ID is possible. terminal sets contain a list of fully and partially qualified terminal names.

user

Each user is represented by a user ID. The term "user" refers to persons, applications, procedures etc. that may be granted access to the operating system and thus to the computer via a user ID.

user administration

All those user IDs of a DP system which are authorized to regulate the allocation of resources and the assignment of user rights to user IDs and user groups and to create, modify and delete user IDs and user groups. They include the group administrators as well as global user administration.

user attributes

All the characteristic features of a user ID which are stored in the user catalog.

user command

Command which may be issued under any user ID either in system mode (/) or in program mode by means of a CMD macro.

user group

Consists of one or more user IDs. Each user group is assigned a name (group ID).

user ID

Name of up to 8 characters entered in the user catalog. The user ID is used for identification for system access. The files and job variables managed by the operating system are assigned to a particular user ID. The assignment is recorded in the file directory.

user ID catalog

The file $TSOS.SYSSRPM which contains the user attributes of all user IDs of a pubset.
Synonym: user catalog

user organization

The organization of user IDs in user groups. It permits both the emulation of existing organizational structures and the project-oriented grouping of users.

user privilege

All those attributes assigned to a user ID and stored in the user ID catalog that convey rights.

4 Related publications

You will find the manuals on the internet at http://manuals.ts.fujitsu.com.

[1] **SECOS**

Security Control System - Access Control

User Guide

[2] **BS2000 OSD/BC**

Introduction to System Administration

User Guide

[3] **BS2000 OSD/BC**

System Installation

User Guide

[4] **BS2000 OSD/BC**

Commands

User Guide

[5] **ARCHIVE** (BS2000)

User Guide

[6] **BS2000 OSD/BC**

Introductory Guide to DMS

User Guide

[7] **BS2000 OSD/BC**

DMS Macros

User Guide

[8] **EDT** (BS2000)

Statements

User Guide

[9] **FDDRL** (BS2000)

User Guide

[10] **openFT for BS2000**

Enterprise File Transfer in the Open World

User Guide

[11] **FTAC-BS2000** (TRANSDATA)

Extended Access Control for File Transfer

User’s Guide

[12] **HSMS** (BS2000)

Hierarchical Storage Management System

Volume 1: Functions, Management and Installation

User Guide

[13] **HSMS** (BS2000)

Hierarchical Storage Management System

Volume 2: Statements

User Guide

[14] **BS2000 OSD/BC**

Utility Routines

User Guide

[15] **BS2000 OSD/BC**

Executive Macros

User Guide

[16] **MAREN** (BS2000)

Tape Management in BS2000

User Guide

[17] **openUTM** (BS2000, UNIX, Windows)

Generating Applications

User Guide

[18] **BS2000 OSD/BC**

System Exits

User Guide

[19] **SDF** (BS2000)

SDF Dialog Interface

User Guide

[20] **openSM2** (BS2000)

Software Monitor

Volume 1: Administration and Operation

[21] **VM2000**

Virtual Machine System

User Guide

[22] **LMS** (BS2000)

SDF Format

User Guide

[23] **SDF-P** (BS2000)

Programming in the Command Language

User Guide

[24] **POSIX** (BS2000)

POSIX Basics for Users and System Administrators

User Guide

[25] **POSIX** (BS2000)

Commands

User Guide

[26] **C Library Functions** (BS2000)

for POSIX Applications

Reference Manual

[27] **SPOOL** (BS2000)

User Guide

[28] **SPOOL** (BS2000)

Part 2, Utility Routines

User Guide

[29] **BS2000 OSD/BC**

Migration Guide

User Guide

[30] **PROP-XT** (BS2000)

Programmed Operating with SDF-P

Product Manual

[31] **JV** (BS2000)

Job Variables

User Guide

[32] **BS2000 OSD/BC**

System-Managed Storage

User Guide

[33] **SESAM/SQL-Server** (BS2000)

Database Operation

User Guide

Other publications

This publication cannot be obtained from Fujitsu Technology Systems.

[34] **IT Security Criteria**

Criteria for the Evaluation of Trustworthiness of Information Technology (IT) Systems

(published by GISA - German Information Security Agency on behalf of the Government of

the Federal Republic of Germany)

1st Version of 11 January, 1989

Cologne, Bundesanzeiger, 1989

ISBN 3-88784-200-6