SECOM Product Overview

What is Secure Command Manager (SECOM)?

SECOM is a command management and session control tool. It enforces a sharp differentiation between:

- functional users (generic user IDs, e.g. SUPER.SUPER), and
- individual users (real people, e.g. GHS.CARL, Alias Weber_Carl)

and provides a method for separation and segregation of duties.

The Problem

Normally, individual users must logon to a function to perform tasks on behalf of the function. For example:

- System wide backup, where the operator needs READ access to all files, but just for backup tasks
- Application management (e.g. PATHCOM access to $APPL)
- Database management (e.g. SQLCI, FUP)
- System management (e.g. start-up at cold load time, executing OBEY files or TACL macros)

Logging on to a function has several drawbacks:

- The password has to be known, and it can be used any time to get access to the function.
- Because a password allows a logon from scratch, real auditing cannot be enforced.
- The password can be given to 'anybody' - without any trace - and misused.
- Logging on to a function means having access to all the resources the function has access to.

The only Tandem based solution available today to address these types of problems is to use PROGIDed programs, where programs have a SAFEGUARD ACL to protect them against misuse. But the management of PROGIDed software is itself problematic, especially when a GUARDIAN release change has to be performed.
An additional point for concern is that accessing a program (FUP, SQLCI) means having access to all of the program's functions. For example: to 'UP' a disk volume, the program PUP must be started with a SUPER-group ID. But running PUP with a SUPER-group ID also allows the user to 'DOWN' a disk.

The SECOM Solution

SECOM provides THE solution. You can quickly and effectively administer command management and session control across a network from a single system.

Features

- SECOM can run in four execution modes:
  1. BATCH mode: The SECOM command is delivered in the start-up message (like: FUP INFO *). For example:

     ```
     SECOM SHUTDOWN $APPL1
     ```

  2. OBEY mode: SECOM reads and executes SECOM commands from an IN file. For example:

     ```
     SECOM/IN OBEY/
     ```

  3. INTERACTIVE mode: SECOM prompts the user for a SECOM command. For example:

     ```
     $DSMSCM SECOM600 41> secom
     SECOM (604) - T7172G06.AFV - (30Nov2000)   System \BEECH, running NSK G06
     Copyright (c) GreenHouse Software & Consulting 1995-2000
     Running in CPU 1 at PRI 158
     Invoking \BEECH.$GHS1.SECOM.GHSCSTM (Section SECOM)
     SECOM is running local
     LazyTyping set to: ABBREVIATED
     <S>
     ```

  4. INLINE mode:

     ```
     $GHS1 SECOM 36> secom/INLINE/
     ```

- A SECOM command is a free format, non case-sensitive string up to 32 bytes long. A SECOM command can be attached to a function key.
- Each SECOM command can have a description line, which is displayed at execution time. If the description line is displayed, the SECOM user is also requested to confirm the command by typing a 'Y'.

  In addition, every SECOM command may have extended help, featuring up to a page of information.
- Five SECOM command types are supported:
  1. Batch: SECOM command attributes can NOT be modified by the user at run time.
  2. Interactive: SECOM command attributes CAN be modified by the user at run time. For example CPU, Name, PRI, IN, OUT, SWAP etc.
  3. Macro: A TACL MACRO or TACL ROUTINE is executed.
  4. Checked: The control is done on the command level; the number of checked commands is unlimited; wild cards are supported. Checked commands can be executed as batch and interactive commands.
  5. Concatenated: A set of SECOM commands is executed, where the list and sequence is maintained within SECOM.
- Each SECOM command runs at a pre-defined (functional) user ID. This ID does not need to exist; SECOM is able to execute resources on behalf of non existing IDs.

- Each SECOM command can have an unlimited number of 'real' users attached to it.
  - User groups are supported
    - the number of users in a group is unlimited
    - up to three group stacks are supported (a group can have an entry in a group etc.)
  - Starting with D30, Alias user names are supported (up to 32 bytes, case sensitive)
  - GUARDIAN as well as Alias names support wildcards (* and ? characters)
  - Additional authentication per user by
    - system password, or a
    - token based Challenge/Response handshake
  - Optional command witnessing (n out of m users have to witness the execution of a command)
    This as well allows a remote authorization
  - DENY flag per user/user group entry
  - Grouping of SECOM commands, e.g.
    - set of command for the EMERGENCY case
    - set of command to manage APPLICATION1
- Each SECOM command can have time windows attached to it. The number of windows is unlimited. Each time window may be based on a week day, and time, e.g. MON 10:00 - 12:00.
- The execution of SECOM commands can be restricted to a set of given
  - static terminals
  - IP addresses
  - X.25 DTE numbers
- Each SECOM command is mapped to a 'real' program. This program can be pre-defined with all attributes (IN, OUT, PRI, Nowait, LIB etc.).
- Each SECOM command may have a start-up message.
- Each SECOM command may require additional witnessing by a predetermined number of witnesses.
- The delivery of PARAM and ASSIGN messages is supported.
- SECOM inherits DEFINES, and supports loading/deleting of DEFINES.
- When in CHECKED mode, SECOM checks the command a user provides against a set of pre-defined commands. The following parameters apply:
  - A set of commands can be allowed or denied (ALL_BUT modifier).
  - Support for wild cards (*, ? as well as &).
  - Support for [T] (template) with wild cards.
- All SECOM activities are logged (audited).
- EMS event messages can be generated on a SECOM command basis.
- SECOM is able to enforce two types of terminal I/O trace:
  - only the user's input is captured and logged
  - the user's terminal input and the system output is logged
- SECOM commands and a SECOM session may have an inactivity timeout defined for it, that can include a session clean-up.

- SECOM commands can get functional attributes (for example EMERGENCY, and OPERATIONS) and users and user groups can be attached to these functions.
- SECOM directs new processes in:
  - a pre-defined CPU, or
  - the least busy CPU (based on a CPU mask, and an actual measurement), or
  - all CPUs using a round robin method (based on a CPU mask)
- The SECOM user can customize his SECOM commands, and environment.

SECOM's Development and Run-Time Environment

- SECOM is developed and tested on
  - Himalaya (K122)
  - Integrity (S7000) and
  - Itanium (NS1000)
  type systems
- SECOM runs on GUARDIAN D40 or better
- SECOM is based on ENSCRIBE type files
- SECOM uses basic GUARDIAN functions (Not PATHWAY, SAFEGUARD, SQL, TMF etc.); this allows it to run even directly after a cold load
- SECOM obeys GUARDIAN and SAFEGUARD security rules
- SECOM runs PRIV code !
- SECOM supports all GUARDIAN-Dxx features. It
  - runs at HighPIN
  - accepts HighRequesters
  - starts resources with HighPIN set to ON
  - uses long process names (configurable)
- The SECOM database is maintained by a PATHWAY application.
  A windows based GUI is available as well.
- SECOM is Y2000 compliant.

Using basic GUARDIAN functions enables SECOM to run on a cold loaded system.

Subsystems

SECOM comes with the following subsystems:

- SECMAN
  This is an interactive system to manage the data base of SECOM 'by hand'. It as well functions as service, allowing a central management of the SECOM environment, running on remote Tandem systems.
  SECMAN offers an efficient LIST capability of SECOM access rights.
- SECOMCI block mode interface
  Executing a SECOM command by simply pressing a function key.
- Challenge/Response Device Maintenance
  Supports the Atalla Challenge/Response Device and devices compatible with it.
- PATHWAY application $SECMAIN to maintain the SECOM data base
  Supports OBJECTTYPE like management authority, and a complete management log.

- 'Black Hole' NonStop process, acting like a 'waste basket' to dispose of any output produced by a program (Dxx version of $NULL process).
- Terminal I/O tracer (input as well as input and output trace).
- Process Control Program, allowing process control (e.g. stopping a process).
- PATHWAY application $SECMAIN to maintain the SECOM database
  Supports OBJECTTYPE like management authority, and a complete management log.

Optional Sub-systems

- Atalla Challenge/Response Device
  Required for Challenge/Response authentication.
- Authentication Server $AS
  Adds TimeToken authentication and allows SECOM to run as initial command interpreter (ICI).
- OSS enhanced tracer
  allows a terminal trace of an OSH session.GreenHouse Administration Suite, a Browser based application to manage the SECOM data base.Trace and Log Information System TALIS to evaluate log and trace files
- A variety of supportive utilities.

Delivery

- SECOM program
- DDL Source
- All non-optional subsystems
- Challenge/Response maintenance programs (INITACR, CRDCOM)
- Documentation (in English)
- Pre-defined ENFORM queries to list the SECOM-LOG and TRACER information
- $SECMAIN PATHWAY application, featuring OBJECTTYPE level security

Escrow Agent

GHS is willing to put all sources into escrow.

Availability

A fully functional version of SECOM is available for free for a 2 month trial.