LAYER SEVEN SECURITY

SIEM INTEGRATION FOR SAP©

SAP-SIEM INTEGRATION FOR ADVANCED THREAT DETECTION

WHITE PAPER

SIEM INTEGRATION FOR SAP

SAP-SIEM INTEGRATION FOR ADVANCED THREAT DETECTION

CONTENTS

SECTION 1 SUMMARY

Security Information and Event Management (SIEM) platforms combine the ability to collect log data from applications, hosts, routers, switches, firewalls and other endpoints with the ability to analyze events to support threat detection, event correlation and incident response.

SIEM platforms require complete coverage for maximum yield. In other words, organizations reap the full benefits of SIEM platforms when monitoring logs throughout the technological infrastructure. This includes SAP application logs for organizations with SAP systems.

However, there are several challenges with integrating SAP application logs with SIEM systems. The first challenge is complexity. SAP systems typically contain multiple logs that capture security-relevant events. The SAP NetWeaver Application Server ABAP (AS ABAP) alone has at least seven such logs including the Security Audit Log, Gateway Server Log, HTTP Log, System Log, Transaction Log, Change Document Log, and the Read Access Log. The logs do not have a standardized format or structure. Some are captured at the file level and others are stored in SAP tables. The complexities involved in integrating multiple and distinct logs from each SAP system should not be underestimated, especially for large SAP landscapes.

The second challenge is log volume. Raw event logs can grow to gigabytes and even terabytes within a relatively short period of time in SAP systems that often support thousands of end users and hundreds of cross-system connections. Transmitting large volumes of log data from SAP systems to SIEM platforms could consume high levels of network bandwidth. The need to store such data for analysis could also increase resource requirements and licensing costs for SIEM systems.

The third challenge with directly integrating SAP logs is maintenance. Monitoring and supporting the numerous integration points between SAP systems and SIEM platforms, as well as regular archiving to deal with the accumulation of log data, could lead to high maintenance costs.

Finally, many SAP logs do not natively include information to support cross-platform correlation using SIEM tools. This includes source and destination IPs for security events. Values for sources and destinations in SAP logs are often terminal names and SAP System IDs (SIDs) rather than IP addresses. Therefore, Security Operations Centers (SOCs) are not able to easily correlate SAP events with non-SAP events in SIEM platforms.

The challenges of log complexity, volume, maintenance, and correlation can be overcome by monitoring SAP event logs with the Cybersecurity Extension for SAP. The Cybersecurity Extension for SAP is an SAP-certified cybersecurity platform designed to protect SAP landscapes through threat detection, vulnerability management, patch management, compliance monitoring, custom code security, and access risk analysis.

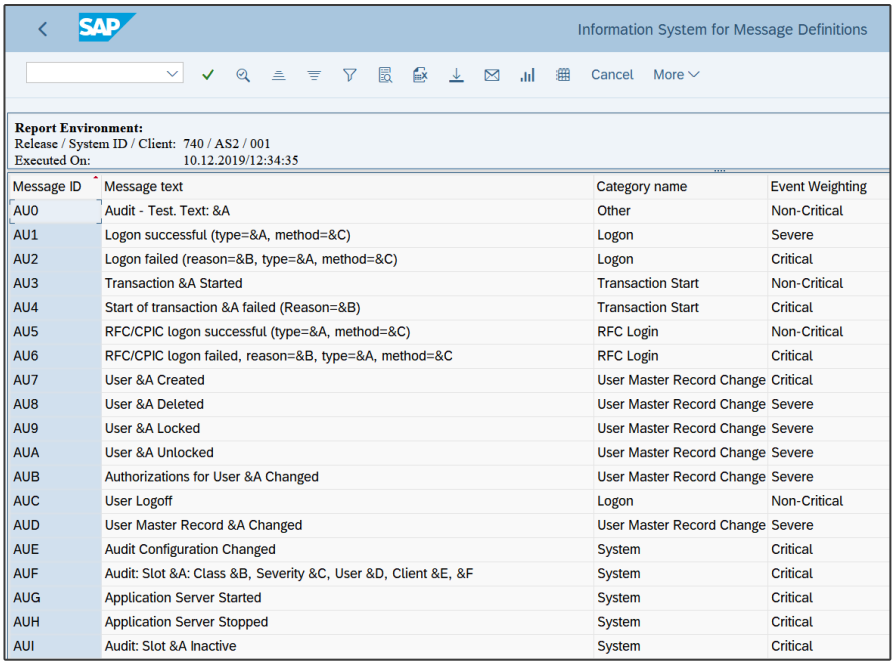
The monitoring and alerting capabilities in the Cybersecurity Extension for SAP connect directly to SAP event logs to detect indicators of compromise and trigger alerts for security events. Alerts can be enriched, filtered, normalized, and forwarded to SIEM platforms, providing a simpler, easier, and faster method for integrating SAP security events with enterprise monitoring solutions.

SECTION 2 SAP LOGS

SECURITY AUDIT LOG

The Security Audit Log is the main source of security-related event information in the SAP NetWeaver Application Server ABAP (AS ABAP). Events captured by the log are assigned a unique message ID. Each message ID is categorized in an audit and event class. Audit classes include dialog logons, RFC/ CPIC logons, RFC calls, transaction starts, report starts, user master changes, and system events. Event classes include critical, severe and non-critical. Supported message IDs can vary between systems based on the version of AS ABAP. Report RSAU_INFO_SYAG displays available message IDs in each system.

Figure 2.1 Report RSAU_INFO_SYAG



Report Environment:
Release / System ID / Client: 740 / AS2 / 001
Executed On: 10.12.2019/12:34:35

Message ID	Message text	Category name	Event Weighting
AU0	Audit - Test. Text: &A	Other	Non-Critical
AU1	Logon successful (type=&A, method=&C)	Logon	Severe
AU2	Logon failed (reason=&B, type=&A, method=&C)	Logon	Critical
AU3	Transaction &A Started	Transaction Start	Non-Critical
AU4	Start of transaction &A failed (Reason=&B)	Transaction Start	Critical
AU5	RFC/CPIC logon successful (type=&A, method=&C)	RFC Login	Non-Critical
AU6	RFC/CPIC logon failed, reason=&B, type=&A, method=&C	RFC Login	Critical
AU7	User &A Created	User Master Record Change	Critical
AU8	User &A Deleted	User Master Record Change	Severe
AU9	User &A Locked	User Master Record Change	Severe
AUA	User &A Unlocked	User Master Record Change	Severe
AUB	Authorizations for User &A Changed	User Master Record Change	Severe
AUC	User Logoff	Logon	Non-Critical
AUD	User Master Record &A Changed	User Master Record Change	Severe
AUE	Audit Configuration Changed	System	Critical
AUF	Audit: Slot &A: Class &B, Severity &C, User &D, Client &E, &F	System	Critical
AUG	Application Server Started	System	Critical
AUH	Application Server Stopped	System	Critical
AUI	Audit: Slot &A Inactive	System	Critical

Audit classes, event classes and message IDs are selected for auditing using filters maintained in transaction SM19. Recommended settings include all actions covering all audit classes and events in every client for standard users such as SAP* (see 2.2), severe and critical events, and the specific non-critical events specified in fig. 2.3.

Figure 2.2 Filter Settings for Auditing Actions Performed by SAP* User

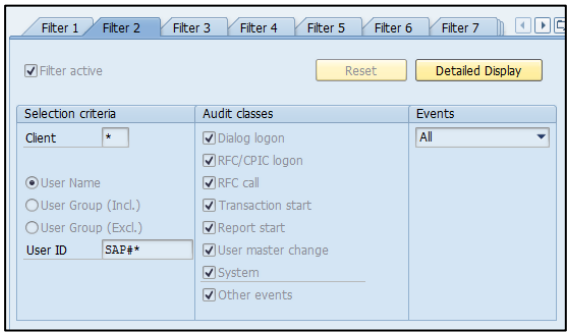


Figure 2.3 Recommended Non-Critical Events for Auditing

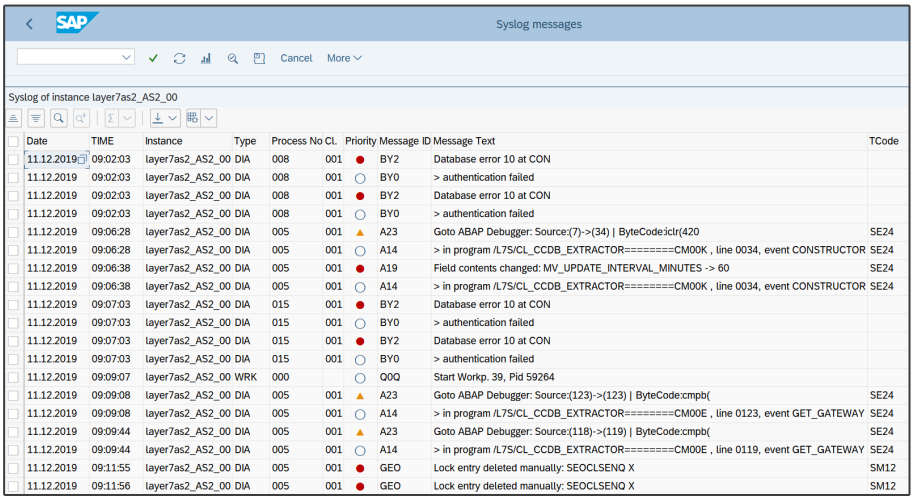
MESSAGE ID	EVENT
BU4	Dynamic ABAP Code
CUY	Debugging Users
DU9	Generic Table Access
DUI	RFC Callback Executions
FU1	RFC Calls with Dynamic Destinations
AU5	RFC/CPIC Logon
AUK	RFC Function Call
AUW	Report Start
EUF	Failed RFC Function module Call
EUG	Failed RFC Function module Call
EU5	Deletion of Audit Log Data
BU2	Password Change

The security audit log is not activated by default and should be enabled using the setting 1 for profile parameter rsau/enable. The maximum size of the audit log should be adjusted to higher than the default value of 1,000,000 bytes using parameter rsau/max_diskspace_local. The log is stored as a UTF-16LE encoded file stored in the directory specified by the rsau/local/file parameter. Log events can be read and backed up using transaction SM20 and deleted using transaction SM18.

SYSTEM LOG

System-related errors and warnings in AS ABAP are logged in the System Log. This includes events such as debugging, user locks/ unlocks, logon attempts for locked users, and changes to audit settings. Unique message IDs are used for each event. The System Log can be displayed using transaction SM21.

Figure 2.4 System Log



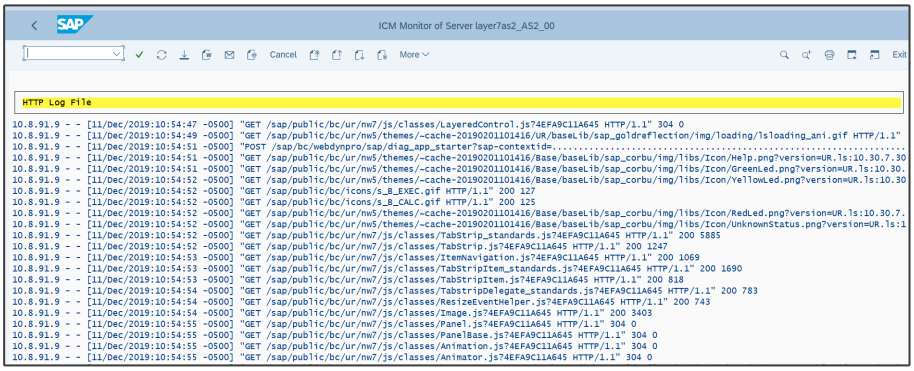
Date	TIME	Instance	Type	Process No	Cl.	Priority	Message ID	Message Text	TCode
11.12.2019	09:02:03	layer7as2_AS2_00	DIA	008	001	●	BY2	Database error 10 at CON	
11.12.2019	09:02:03	layer7as2_AS2_00	DIA	008	001	○	BY0	> authentication failed	
11.12.2019	09:02:03	layer7as2_AS2_00	DIA	008	001	●	BY2	Database error 10 at CON	
11.12.2019	09:02:03	layer7as2_AS2_00	DIA	008	001	○	BY0	> authentication failed	
11.12.2019	09:06:28	layer7as2_AS2_00	DIA	005	001	▲	A23	Goto ABAP Debugger: Source:(7)->(34) | ByteCode:iclr(420	SE24
11.12.2019	09:06:28	layer7as2_AS2_00	DIA	005	001	○	A14	> in program /L7S/CL_CCDB_EXTRACTOR========CM00K , line 0034, event CONSTRUCTOR	SE24
11.12.2019	09:06:38	layer7as2_AS2_00	DIA	005	001	●	A19	Field contents changed: MV_UPDATE_INTERVAL_MINUTES -> 60	SE24
11.12.2019	09:06:38	layer7as2_AS2_00	DIA	005	001	○	A14	> in program /L7S/CL_CCDB_EXTRACTOR========CM00K , line 0034, event CONSTRUCTOR	SE24
11.12.2019	09:07:03	layer7as2_AS2_00	DIA	015	001	●	BY2	Database error 10 at CON	
11.12.2019	09:07:03	layer7as2_AS2_00	DIA	015	001	○	BY0	> authentication failed	
11.12.2019	09:07:03	layer7as2_AS2_00	DIA	015	001	●	BY2	Database error 10 at CON	
11.12.2019	09:07:03	layer7as2_AS2_00	DIA	015	001	○	BY0	> authentication failed	
11.12.2019	09:09:07	layer7as2_AS2_00	WRK	000		○	Q0Q	Start Workp. 39, Pid 59264	
11.12.2019	09:09:08	layer7as2_AS2_00	DIA	005	001	▲	A23	Goto ABAP Debugger: Source:(123)->(123) | ByteCode:cmpb(	SE24
11.12.2019	09:09:08	layer7as2_AS2_00	DIA	005	001	○	A14	> in program /L7S/CL_CCDB_EXTRACTOR========CM00E , line 0123, event GET_GATEWAY	SE24
11.12.2019	09:09:44	layer7as2_AS2_00	DIA	005	001	▲	A23	Goto ABAP Debugger: Source:(118)->(119) | ByteCode:cmpb(	SE24
11.12.2019	09:09:44	layer7as2_AS2_00	DIA	005	001	○	A14	> in program /L7S/CL_CCDB_EXTRACTOR========CM00E , line 0119, event GET_GATEWAY	SE24
11.12.2019	09:11:55	layer7as2_AS2_00	DIA	005	001	●	GEO	Lock entry deleted manually: SEOCLSENQ X	SM12
11.12.2019	09:11:56	layer7as2_AS2_00	DIA	005	001	●	GEO	Lock entry deleted manually: SEOCLSENQ X	SM12

The System Log is enabled by default and does not require any specific maintenance tasks since the log is a ring buffer that automatically overwrites the oldest data. The maximum size of the system log is specified by the parameter rslg/max_diskspace/local. The default value of the parameter is 500,000 bytes. Central system logs can be configured for Linux platforms. Application servers send local logs to the central server. The central log is comprised of an active and inactive file. The current log is stored in the active file. A log switch is performed when the maximum size of the active file is reached, and a new inactive log file is automatically created. The switch occurs when the size of the active log file is half the value as specified in the rslg/max_diskspace_central parameter. The default value for the parameter is 2,000,000 bytes. The location of the local log is specified in the rslg/local/file profile parameter. The location of the active file for the central System log is specified in the rslg/central/file profile parameter and the location of the inactive file is specified in rslg/central/old_file.

ICM LOG

Web-based communication including HTTP(S) and SMTP calls to AS ABAP are captured in the log for the Internet Communication Manager (ICM). The profile parameter icm/HTTP/logging_<xx> controls the logging for inbound requests and icm/HTTP/logging_client_<xx> regulates logging for outbound requests. <xx> specifies the port reference for the supported protocols. For example, the relevant parameter for HTTP logging would be icm/HTTP/logging_0 if icm/server_port_0 = PROT=HTTP. The syntax for the ICM logging parameters includes options for the log file format. The CLF format is recommended since this option supports logging of URLs in log entries. The ICM will log access to dangerous URLs including URLs for ICF services with known security vulnerabilities. This includes IDOC MXL, SOAP RFC and WEBRFC. Client and server ICM logs can be displayed using transaction SMICM or directly in the work directory of each instance.

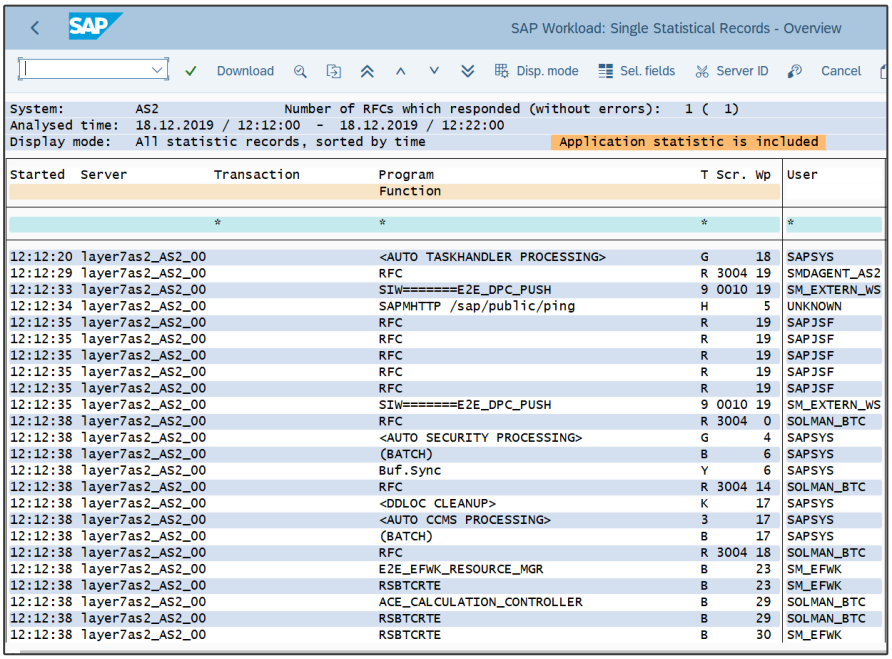
Figure 2.5 ICM Log



BUSINESS TRANSACTION ANALYSIS

Statistical records can be monitored to identify calls for dangerous transactions and programs including areas such as system administration, user maintenance, transports, and RFC administration. The records can be viewed using transaction STAD. Statistical records are stored chronologically in a buffer in each application server. The buffer is flushed to a statistics file when it is full. A new file is created each hour. The oldest file in the directory is automatically deleted every hour. The default directory for the files is /usr/sap/<SysID>/<Instance Directory>/DATA. The maximum number of statistic files is determined by the value of the profile parameter stat/max_files. The default value for the parameter is 48. Therefore, STAD data is only available for 48 hours. The maximum value for the parameter is 99. This would retain STAD data for 4 days.

Figure 2.6 STAD



GATEWAY LOG

The gateway log supports monitoring for RFC communications. This includes actions such as opening and closing of network connections, monitor commands, registration and deregistration of servers, and launching of external programs. Logging for gateway actions is configured by maintaining the relevant indicators for each action using the ACTION option in the profile parameter gw/logging. The default setting for the parameter only logs changes to security settings and rejected actions. The gateway log is stored in a file located in the work directory of each SAP instance. The prefix for each entry in the log denotes the gateway action.

Figure 2.7 Gateway Log

```
S Sun Dec 18 2016 11:11:16:442  P USER=* USER-HOST=local HOST=internal TP=*
S Sun Dec 18 2019 11:11:16:442  P USER=* USER-HOST=internal HOST=local TP=*
S Sun Dec 18 2019 11:11:16:442 (re)load reginfo file C:\usr\sap\AS2\SYS\global\reginfo.DAT, version=2 (3 lines, mode=1)
S Sun Dec 18 2019 11:11:16:442  P TP=* HOST=local
S Sun Dec 18 2019 11:11:16:442  P TP=* HOST=internal
S Sun Dec 18 2019 11:11:16:442 prxyinfo file C:\usr\sap\AS2\DVEBMGS00\data\prxyinfo.DAT not found
X Sun Dec 18 2019 11:24:06:969 gateway stopped, pid=2628
P Sun Dec 18 2019 11:24:06:969 log file closed
P Sun Dec 18 2019 11:24:22:483 log file reopened
P Sun Dec 18 2019 11:24:22:483 initial gw/logging ACTION=Ss LOGFILE=gw_log-%y-%m-%d SWITCHTF=day MAXSIZEKB=100
X Sun Dec 18 2019 11:24:22:483 gateway started, pid=4120
P Sun Dec 18 2019 11:24:22:483 gw/logging = ACTION=Ss LOGFILE=gw_log-%y-%m-%d SWITCHTF=day MAXSIZEKB=100
S Sun Dec 18 2019 11:24:22:483 simulation mode deactivated
S Sun Dec 18 2019 11:24:22:483 gw/reg_no_conn_info: 1
S Sun Dec 18 2019 11:24:22:487 (re)load secinfo file C:\usr\sap\AS2\SYS\global\secinfo.DAT, version=2 (5 lines, mode=1)
S Sun Dec 18 2019 11:24:22:487  P USER=* USER-HOST=local HOST=local TP=*
S Sun Dec 18 2019 11:24:22:487  P USER=* USER-HOST=local HOST=internal TP=*
S Sun Dec 18 2019 11:24:22:487  P USER=* USER-HOST=internal HOST=local TP=*
S Sun Dec 18 2019 11:24:22:487 (re)load reginfo file C:\usr\sap\AS2\SYS\global\reginfo.DAT, version=2 (3 lines, mode=1)
S Sun Dec 18 2019 11:24:22:487  P TP=* HOST=local
S Sun Dec 18 2019 11:24:22:487  P TP=* HOST=internal
S Sun Dec 18 2019 11:24:22:487 prxyinfo file C:\usr\sap\AS2\DVEBMGS00\data\prxyinfo.DAT not found
X Sun Dec 18 2019 16:05:46:083 gateway stopped, pid=4120
P Sun Dec 18 2019 16:05:46:083 log file closed
P Sun Dec 18 2019 16:17:23:719 log file reopened
P Sun Dec 18 2019 16:17:23:719 initial gw/logging ACTION=Ss LOGFILE=gw_log-%y-%m-%d SWITCHTF=day MAXSIZEKB=100
X Sun Dec 18 2019 16:17:23:719 gateway started, pid=3636
P Sun Dec 18 2019 16:17:23:719 gw/logging = ACTION=Ss LOGFILE=gw_log-%y-%m-%d SWITCHTF=day MAXSIZEKB=100
```

CHANGE DOCUMENTS

Change documents support auditing for changes to critical objects. Objects consist of one or more table. User-related changes are logged in change documents within the IDENTITY and PFCG object class. This includes adding/ removing users and profiles. The change document header in table CDHDR logs changes for all objects and classes. However, the complete details of change documents can be viewed in table CDPOS. This includes old and new values. Each row in CDPOS includes a change flag. This field will include the value U for updates, I for insert, and D for delete.

READ ACCESS LOGGING

Read Access Logging (RAL) monitors access to sensitive data based on customer-specific scenarios configured with SRALMANAGER. This can include Personally Identifiable Information (PII) such as social security numbers, credit card numbers, and banking information. RAL supports monitoring for RFC, dynpro, web dynpro and web service channels. It can log access to sensitive data at both the field and table level. RAL entries can be viewed using transaction SRALMONITOR or in table SRAL_LOG. RAL is archived using archiving object SRAL with transaction SARA.

Figure 2.8 Read Access Log

Created At (Local Ti...	User Name	System ID	Channel	Direction	Logging Purpose	Client IP Address
05.08.2019 06:26:15,00	ATTACKER	AS2	Dynpro	Output	SE16_USR02	10.8.91.4
05.08.2019 06:26:15,00	ATTACKER	AS2	Dynpro	Output	TABLE_USER02_ACCESS	10.8.91.4
05.08.2019 02:45:27,00	ATTACKER	AS2	Dynpro	Output	SE16_USR02	10.8.91.4
05.08.2019 02:45:27,00	ATTACKER	AS2	Dynpro	Output	TABLE_USER02_ACCESS	10.8.91.4
05.08.2019 02:45:24,00	ATTACKER	AS2	Dynpro	Output	SE16_USR02	10.8.91.4
05.08.2019 02:45:24,00	ATTACKER	AS2	Dynpro	Output	TABLE_USER02_ACCESS	10.8.91.4
23.07.2019 13:23:11,00	ATTACKER	AS2	Dynpro	Output	SE16_USR02	10.8.91.4
23.07.2019 13:23:11,00	ATTACKER	AS2	Dynpro	Output	TABLE_USER02_ACCESS	10.8.91.4
23.07.2019 12:06:07,00	ATTACKER	AS2	Dynpro	Output	TABLE_USH02_ACCESS	10.8.91.4
23.07.2019 11:53:44,00	ATTACKER	AS2	Dynpro	Output	SE16_USR02	10.8.91.4
23.07.2019 11:53:44,00	ATTACKER	AS2	Dynpro	Output	TABLE_USER02_ACCESS	10.8.91.4

JAVA SECURITY LOG

The security audit log is enabled by default in the SAP NetWeaver Application Server (AS) Java. It logs security-relevant events in Java platforms including successful and unsuccessful logons, user creation, and changes to users, role, groups, and audit or UME properties. The log is stored in the file system within the directory \usr \sap \ <SID> \ <instance_number> \j2ee \cluster \server <n> \log \system. Logs are written to five files. The maximum size of each file is 10 MB. When the fifth file reaches the maximum permitted size, the contents of the oldest file is overwritten. If you enable the archiving process, the set of files is converted into a single ZIP file and stored as an archive on the file system.

Entries in the log follow the format below.
[Timestamp] | [Event Name] | [Event Type] | [ObjectID] | [ObjectName] | [Details]
The UME properties ume.secaudit.log_actor, ume.secaudit.get_object_name, and ume.logon.security_policy.log_client_hostaddress should be set to true in order to log usernames, objects names and IP addresses in audit entries.

Figure 2.9 Java Security Log

```
#2.0#2019 12 18 13:05:31:868#+00#Info#/System/Security/Audit/PrincipalModification#
#BC-JAS-SEC-UME#com.sap.security.core.sda#C000AC1F01BF00210000009000011B8#5334650000001859#sap.
User created    | USER.CREATE   | USER.PRIVATE_DATASOURCE.un:cup_app    |       | SET_ATTRIBUTE:

#2.0#2019 12 18 13:05:31:868#+00#Info#/System/Security/Audit/PrincipalModification#
#BC-JAS-SEC-UME#com.sap.security.core.sda#C000AC1F01BF00210000000A000011B8#5334650000001859#sap.
User account created    | USERACCOUNT.CREATE    | UACC.PRIVATE_DATASOURCE.un:cup_app    |

#2.0#2019 12 18 13:05:31:953#+00#Info#/System/Security/Audit/PrincipalModification#
#BC-JAS-SEC-UME#com.sap.security.core.sda#C000AC1F01BF00210000000B000011B8#5334650000001859#sap.
Role modified   | ROLE.MODIFY   | ROLE.UME_ROLE_PERSISTENCE.un:Administrator    |       | ADD_VA

#2.0#2019 12 18 13:05:44:617#+00#Info#/System/Security/Audit/PrincipalModification#
#BC-JAS-SEC-UME#com.sap.security.core.sda#C000AC1F01BF00120000001200011B8#5334650000000613##com
Role created    | ROLE.CREATE   | ROLE.UME_ROLE_PERSISTENCE.un:view-creator.CTCView     |

#2.0#2019 12 18 13:05:45:356#+00#Info#/System/Security/Audit/PrincipalModification#
#BC-JAS-SEC-UME#com.sap.security.core.sda#C000AC1F01BF00250000000F000011B8#5334650000003106#sap.
Role modified   | ROLE.MODIFY   | ROLE.UME_ROLE_PERSISTENCE.un:SAP_XI_ADMINISTRATOR_J2EE
```

HANA AUDIT LOG

Auditing in the HANA database is enabled by setting the value of the system property global_auditing_state to true. The default value is false. Therefore, auditing is not enabled in the default configuration of SAP HANA databases. The audit log can be written to a table, syslog and/or csv files. Table and syslog are recommended. Hence, the property default_audit_trail_type should include the values SYSLOGPROTOCOL and CSTABLE. For logging to csv files, the value of default_audit_trail_type should include CSVTEXTFILE and the file location can be set using the property default_audit_trail_path. The default file path is /usr/sap/<sid>/<instance>/<host>/trace. The property audit_statement_length should be set to -1 to log complete statements for audit events.

Audited actions are defined in audit policies. Each action corresponds to one or more SQL statement. Policies can be defined for the successful or unsuccessful execution of SQL statements and applied globally for all users or targeted for specific users. Policies can also be targeted for specific objects such as tables, schemas, views or procedures. Audit policies should be configured to log the actions in the table below using the HANA Cockpit or Studio. This includes all actions performed by the standard SYSTEM user, system, role and user changes, and failed logon attempts. The audit level should be set to ALERT or CRITICAL for all actions.

Figure 2.10 Recommended Audit Policy for SAP HANA

ACTION	AUDITED ACTION STATUS	AUDIT LEVEL	USERS
ACTIVATE REPOSITORY CONTENT	SUCCESSFUL	ALERT	
ALL ACTIONS	ALL	CRITICAL	SYSTEM
ALTER PERSISTENCE ENCRYPTION ROOT KEY	SUCCESSFUL	ALERT	
ALTER PERSISTENCE ENCRYPTION	SUCCESSFUL	ALERT	
ALTER PSE	SUCCESSFUL	ALERT	
ALTER STRUCTURED PRIVILEGE	SUCCESSFUL	ALERT	
ALTER USER	SUCCESSFUL	ALERT	
CONNECT	UNSUCCESSFUL	ALERT	
CREATE CERTIFICATE	SUCCESSFUL	ALERT	
CREATE PSE	SUCCESSFUL	ALERT	
CREATE ROLE	SUCCESSFUL	ALERT	
CREATE STRUCTURED PRIVILEGE	SUCCESSFUL	ALERT	
CREATE USER	SUCCESSFUL	ALERT	
DROP CERTIFICATE	SUCCESSFUL	ALERT	
DROP PSE	SUCCESSFUL	ALERT	
DROP ROLE	SUCCESSFUL	ALERT	
DROP STRUCTURED PRIVILEGE	SUCCESSFUL	ALERT	
DROP TABLE	SUCCESSFUL	ALERT	
DROP USER	SUCCESSFUL	ALERT	
EXPORT REPOSITORY CONTENT	SUCCESSFUL	ALERT	
GRANT ANY	SUCCESSFUL	ALERT	
IMPORT REPOSITORY CONTENT	SUCCESSFUL	ALERT	
REVOKE ANY	SUCCESSFUL	ALERT	
SET LICENSE (HANA 1.0) OR SET SYSTEM LICENSE (HANA 2.0)	SUCCESSFUL	ALERT	
SYSTEM CONFIGURATION CHANGE	SUCCESSFUL	CRITICAL	
UNSET LICENSE (HANA 1.0) OR UNSET SYSTEM LICENSE (HANA 2.0)	SUCCESSFUL	ALERT	

Database audit trails can be viewed through the system view AUDIT_LOG using the AUDIT OPERATOR or AUDIT ADMIN system privilege. Results can be exported for offline analysis and storage. Once archived, audit logs can be truncated to manage the size of the AUDIT_LOG table.

Figure 2.11 HANA Audit Log

APPLICATION_NAME	APPLICATION_USER_NAME	XS_APPLICATION_USER_NAME	AUDIT_POLICY_NAME	EVENT_STATUS	EVENT_LEVEL	EVENT_ACTION
HDBStudio	Administrator	SYSTEM	MandatoryAuditPolicy	SUCCESSFUL	CRITICAL	CLEAR AUDIT LOG
HDBStudio	Administrator	SYSTEM	DROP USER	SUCCESSFUL	ALERT	DROP USER
HDBStudio	Administrator	SYSTEM	CREATE ROLE	SUCCESSFUL	ALERT	CREATE ROLE
HDBStudio	Administrator	SYSTEM	GRANT ANY	SUCCESSFUL	ALERT	GRANT ROLE
HDBStudio	Administrator	SYSTEM	CREATE USER	SUCCESSFUL	ALERT	CREATE USER
HDBStudio	Administrator	SYSTEM	GRANT ANY	SUCCESSFUL	ALERT	GRANT ROLE
HDBStudio	Administrator	SYSTEM	ALTER USER	SUCCESSFUL	ALERT	ALTER USER
HDBStudio	Administrator	SYSTEM	GRANT ANY	SUCCESSFUL	ALERT	GRANT PRIVILEGE

SAP BTP AUDIT LOG

SAP BTP audit logging provides a centralized framework for recording security, privacy, and configuration-related events across SAP BTP services, applications, subaccounts, global accounts, and Cloud Foundry environments. SAP provides guidance on which audit categories should be used for different event types, but customers are responsible for defining the final logging approach based on their compliance, security, and business requirements. Common categories include audit.data-access for read access to sensitive personal data, such as the user, access channel, date and time, data identifiers, and attributes that were viewed; audit.configuration for changes to personal data, including the user, timestamp, affected data records, and changed attributes; audit.security-events for security-relevant activity such as logons, role changes, and external access, including the event ID, user, time, source IP address, and supporting event details; and audit.data-modification for changes to configuration data, including the user, timestamp, affected data, and old and new values. In SAP BTP Cloud Foundry, audit logs can record activities performed through the cockpit, CLI, APIs, and automation tools, including changes to organizations, spaces, applications, service instances, routes, entitlements, role assignments, and trust configurations. These logs can be accessed through the Audit Log Retrieval API, filtered using API parameters, or reviewed through the Audit Log Viewer for Cloud Foundry. Audit log APIs follow RFC3986 requirements, timestamps are recorded in UTC, and access should be limited to authorized users such as OrgAuditor and SpaceAuditor roles. By default, audit log data is retained for 90 days, so organizations that require longer retention should retrieve and store the logs externally or enable premium audit log retention with a custom retention period. Since the audit log service provides the logging framework, the completeness of the logs depends on individual SAP BTP services and applications sending the correct audit events.

SAP CLOUD CONNECTOR

SAP Cloud Connector should be treated as a separate log source from SAP BTP audit logging. While SAP BTP audit logs provide visibility into events within the BTP platform, subaccounts, services, and Cloud Foundry environments, SAP Cloud Connector maintains its own local log, trace, and audit files within the customer-managed network. These logs provide visibility into connector operations, configuration changes, blocked requests, successful or attempted access between cloud applications and on-premise systems, and communication through the secure tunnel between SAP BTP and internal resources. The Cloud Connector audit log is particularly important because it records access through exposed virtual hosts and resources, as well as administrative changes to Cloud Connector configuration. Since these records are stored separately from SAP BTP audit logs, organizations should collect and retain them independently, integrate them with centralized monitoring or SIEM platforms, and correlate them with SAP BTP audit logs, identity provider events, Cloud Foundry activity, and on-premise SAP system logs to establish end-to-end visibility across hybrid SAP landscapes. BTP services and applications sending the correct audit events.

SAPROUTER LOG

The SAProuter is a network proxy that filters traffic between SAP systems and external networks. It performs a pivotal role in SAP landscapes by filtering SAP traffic using a more granular approach than is possible with conventional network-level firewalls. Logging for the SAProuter is enabled and configured using option -G. The log will capture connections rejected by the SAProuter based on the route permission table configured in the saprouttab. It will also log connections and disconnections between clients and hosts including IP addresses and hostnames.

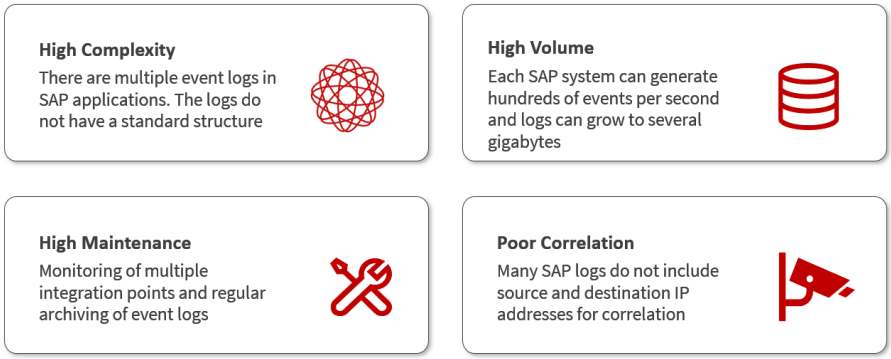
Figure 2.12 SAPRouter Log

```
Wed Dec  4 13:13:59 2019 INIT LOGFILE
Wed Dec  4 13:13:59 2019 READ ROUTTAB ./saprouttab o.k.
Wed Dec  4 13:14:05 2019 CONNECT FROM C1/- host 10.21.72.60/1245 (ldp007.wdf.sap.corp)
Wed Dec  4 13:14:05 2019 CONNECT TO   S1/2 host 10.21.82.77/sapmsBIN (binmain)
Wed Dec  4 13:14:05 2019 DISCONNECT   C1/2 host 10.21.72.60/1245 (ldp007.wdf.sap.corp)
Wed Dec  4 13:14:13 2019 CONNECT FROM C2/- host 127.0.0.1/44997 (localhost)
Wed Dec  4 13:14:13 2019 SEND INFO TO C2/-
Wed Dec  4 13:14:13 2019 DISCONNECT   C2/- host 127.0.0.1/44997 (localhost)
Wed Dec  4 13:14:23 2019 CONNECT FROM C2/- host 10.21.72.60/1276 (ldp007.wdf.sap.corp)
Wed Dec  4 13:14:23 2019 CONNECT TO   S2/1 host 10.21.72.60/3298 (ldp007)
Wed Dec  4 13:14:24 2019 DISCONNECT   S2/1 host 10.21.72.60/3298 (ldp007)
Wed Dec  4 13:14:31 2019 CONNECT FROM C2/- host 10.21.72.60/1352 (ldp007.wdf.sap.corp)
Wed Dec  4 13:14:31 2019 PERM DENIED  C2/- host 10.21.72.60 (ldp007.wdf.sap.corp) to ldp007/23
Wed Dec  4 13:14:31 2019 DISCONNECT   C2/- host 10.21.72.60/1352 (ldp007.wdf.sap.corp)
```

SECTION 3

CYBERSECURITY EXTENSION FOR SAP

The challenges of directly integrating logs from each system in SAP landscapes with SIEM platforms are summarized below. The complexities of integrating multiple logs from numerous systems and managing the various integration points, not to mention the volume of SAP data in SIEM platforms, can lead to long, drawn-out deployments and push up maintenance costs. It may also fail to deliver the desired benefits since SAP event logs often lack the necessary data to support event correlation.



The challenges can be overcome by monitoring SAP event logs using the Cybersecurity Extension for SAP. The solution filters, normalizes, and enriches security event data from SAP logs before forwarding alerts to SIEM systems. The Cybersecurity Extension for SAP monitors SAP logs directly at source without requiring event logs to be extracted and replicated to external repositories, reducing both bandwidth and storage requirements.

The Cybersecurity Extension for SAP supports monitoring across SAP application, database, and host layers, including ABAP, SAP HANA, SAP Java, SAP cloud services such as SAP BTP, and standalone components such as the SAP Cloud Connector, Web Dispatcher, and SAProuter. It periodically analyzes event logs using predefined attack detection patterns to identify indicators of compromise and security-relevant activity. The frequency of monitoring checks is configurable and can be adjusted to meet each organization's security, performance, and operational requirements.

A pattern match triggers the Cybersecurity Extension for SAP to generate alerts for security events and indicators of compromise. Security alerts generated by the solution can be managed through dedicated SAP Fiori applications for alert monitoring, triage, investigation, and reporting. Alerts can also be forwarded to external platforms such as SIEM systems for centralized security monitoring and correlation.

The Cybersecurity Extension for SAP enriches event data with contextual information to support investigation and correlation after ingestion by SIEM platforms. This can include details such as the source system, client, user, event ID, source and destination IP addresses, affected objects, and the activity performed. Event data is normalized using a standardized structure across SAP log sources, helping SIEM platforms process SAP security events consistently.

Alert fields and event details can include values such as alert name, description, date, time, system, system type, priority, user, source and destination IP addresses, and accessed objects such as transactions, reports, function modules, services, or URLs.

A representative alert format could include: <DATE>::<TIME>::<SYSTEM>::<SYSTEM TYPE>::<ALERT TYPE>::<PRIORITY>::<ALERT NAME>::<ALERT DESCRIPTION>::<ALERT DETAILS>
Each value is separated by ::

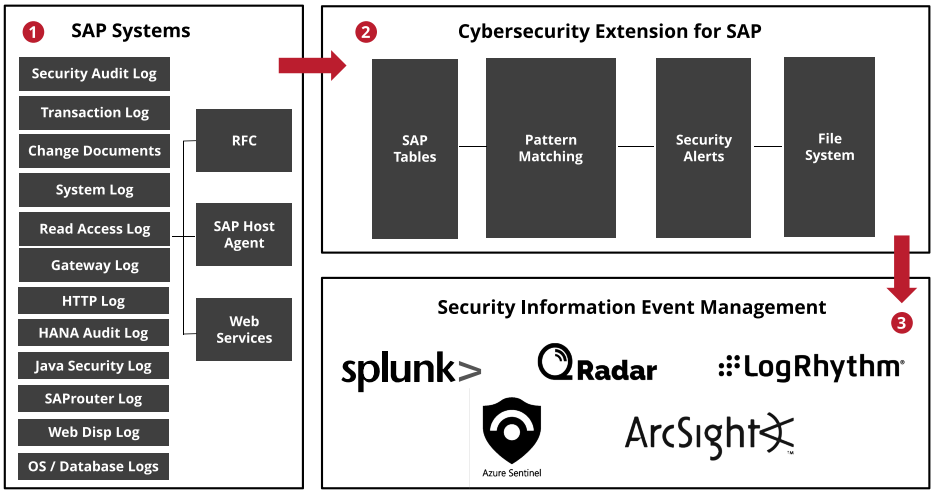
Figure 3.1 Event Structure

AlertDetails <DATE>::<TIME>::<SYSTEM>::<MANAGED OBJECT TYPE>::<ALERT TYPE>::<PRIORITY>::<ALERT NAME>::<ALERT DESCRIPTION>::<ALERT DETAILS>

AlertDetails 20191120::132643::AS2~ABAP::T_SYSTEM::EXCEPTION::8::Generic Table Access::A generic table access tool such as transaction SE16, SE16N, SE17, SM30, SM31, SM34 or SQVI was used to successfully or unsuccessfully access the named table with the executed activity. Select the Guided Procedure for Table Access::SOLMAN_BTC/DU9/184/|DATE:2019-11 20/TIME:08:05:40/ SOURCE_IP:layer7as2.aws.layer7/TARGET_IP:192.168.30.211/CLIENT:001/MSG_TXT: Generic table access call to /SSF/BTAB with activity 03 (auth. check: passed) /TCODE:/REPNA:SAPMSSY1/PRIORITY:Low/PARAM1:/SSF/BTAB/PARAM2:03/PARA

Event files can be stored on the Cybersecurity Extension for SAP host or an external host or file server. A new event file is created by the extension for each day. The contents of the newest file can be periodically pushed to SIEM platforms or pulled by SIEM systems directly from relevant directories. There is a single point of integration for event data between solution and SIEM systems. Hence, maintenance efforts are relatively low.

Figure 3.2 System Architecture



Since event details are written to and stored within alerts in Cybersecurity Extension for SAP, attackers will not be able to remove all traces of malicious actions by modifying event logs alone. They will also need to delete alerts and stop the triggering of email notifications of alerts in the extension. This would be challenging since alerts cannot be deleted in the solution. They can only be confirmed. All alerts are retained and only removed by periodic housekeeping jobs designed to delete aged alerts.

The benefits of SAP-SIEM Integration with SAP Cybersecurity Extension for SAP are summarized below.

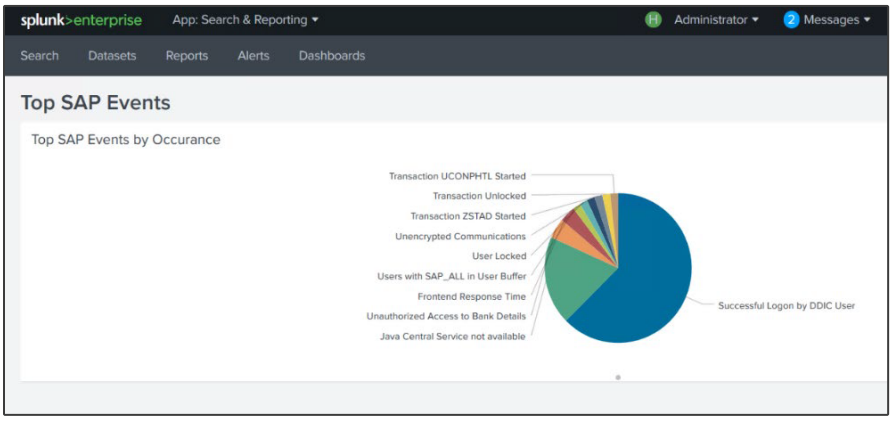
Low Complexity
Automatic coverage for all SAP logs. Event normalization using a standardized structure

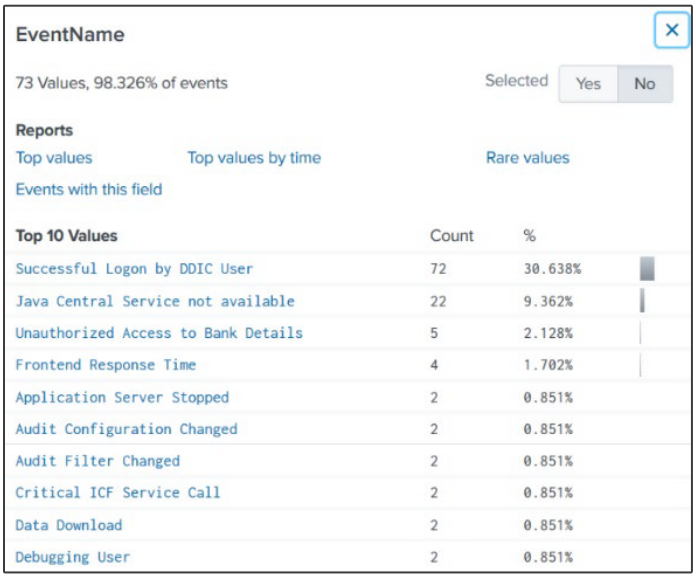
Low Volume
Filtering of SAP events and forwarding of security-relevant events to SIEM

Low Maintenance
Single point of integration between SAP and SIEM

Strong Correlation
Enrichment of event information with source and destination IP addresses

The following illustrates the integration between Cybersecurity Extension for SAP and Splunk Enterprise.





LAYER SEVEN SECURITY

Layer Seven Security is a specialist provider of cybersecurity solutions and services for SAP systems. The company helps organizations protect business-critical SAP solutions from cyber threats, improve compliance readiness, and support secure digital transformations.

Our flagship solution, the Cybersecurity Extension for SAP, is an SAP-certified platform that delivers threat detection, vulnerability management, patch management, compliance monitoring, custom code security, and access risk analysis for SAP landscapes. It integrates with all SIEM products including Splunk and Sentinel.

The solution is designed to support rapid deployment, reduce operational complexity, and provide continuous visibility across SAP solutions and cloud services.

CONTACT US

www.layersevensecurity.com
info@layersevensecurity.com