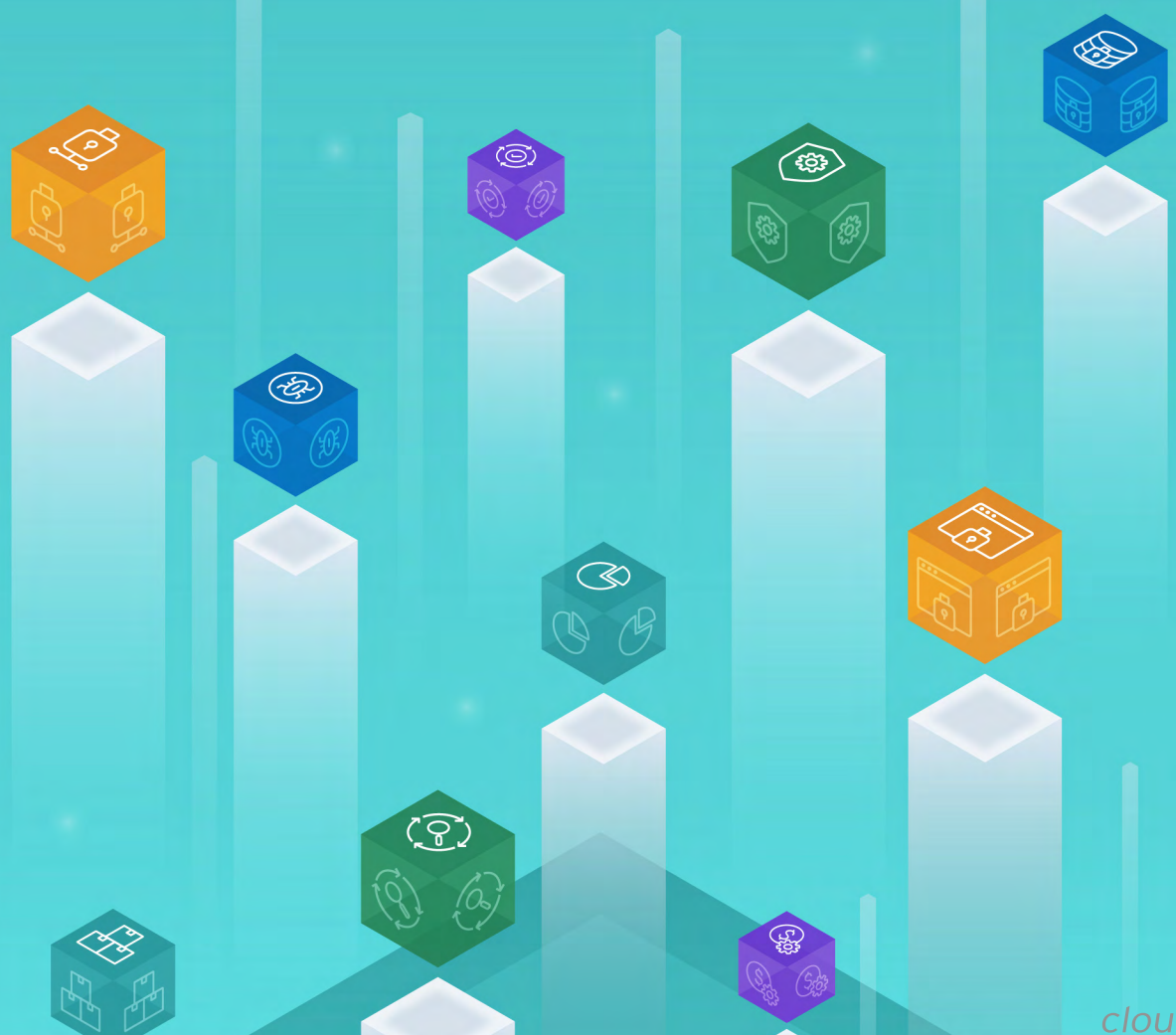
Critical Controls Implementation for SAP (Parts 1 and 2)

Helping Organizations Securely Migrate to and Operate ERP Applications in the Cloud



CSA cloud security alliance®

The permanent and official location for Software Defined Perimeter Working Group is
https://cloudsecurityalliance.org/research/working-groups/enterprise-resource-planning/

Acknowledgments

Lead Authors:

Juan Perez-Etchegoyen
Victor Chin
Sergio Abraham
Hugh Fraser
Thomas Kastner
Fredrik Pihl
Michael Roza
Charlie Singh
Frederik Weidemann

CSA Global Staff:

Victor Chin
Shamun Mahmud
AnnMarie Ulskey (Design)

The Enterprise Resource Planning (ERP) WG seeks to develop best practices to enable organizations that run their business on large ERP implementations, such as SAP or Oracle applications, to securely migrate to and operate in cloud environments. Every ERP deployment is unique to each organization. In most cases organizations spend months if not years customizing their SAP or Oracle implementations and spend a significant amount of money with third-party contractors to complete the implementations. This makes standard security measures more difficult to implement due to the differences of each deployment. With the complexity of these large implementations, combined with the criticality of data and processes housed in these applications, it is imperative that industry best practices be established to provide security guidelines to companies migrating to the cloud in order to protect the organization's critical infrastructure.

Table of Contents

Introduction

The Cloud Security Alliance's Enterprise Resource Planning (ERP) working group aims to help organizations securely migrate to and operate ERP Applications in cloud environments by developing industry best practices. To achieve that goal, the ERP working group developed the Top 20 Critical Controls for Cloud ERP Customers, which was released on June 10, 2019.

At the same time, the ERP working group understands that security configurations and vulnerabilities for cloud ERP applications can be difficult to navigate as there is currently no framework that aligns with standard controls. Furthermore, ERP applications are so complex and diverse that for any guidance document to be truly useful, from an implementation perspective, there is a need to address specific technologies.

The *Critical Controls Implementation for SAP* is the first document in a series of implementation documents the ERP working group hopes to develop that focuses on specific ERP technologies. The first part of the document, released in January 2020, titled *Critical Controls Implementation for SAP (Part 1)*, provided controls implementation guidance for the following controls:

APP01 - Secure Landscape
APP02 - Baseline Secure Configurations
APP03 - Security Vulnerabilities
INT01 – Secure Integrations and API
DAT01 – Continuous Monitoring
DAT02 – Data Separation
DAT03 – Data Encryption
BUS01 - Inventory of Business Assets, Data and Processes
BUS02 - Business Process Controls
BUS03 - Continuous Compliance

With Critical Controls Implementation for SAP (Part 1) released in early 2020, the part 2 of the document was developed by the ERP working group to include the following controls implementation guidance:

USR01 - Secure Authentication
USR02 - User Accounts Management
USR03 - Role-based Access Control
USR04 - Emergency Access
USR05 - Segregation of Duties
USR06 - Secure User Provisioning/Deprovisioning
USR07 – ERP Accounts Security
APP04 - Secure Communications
APP05 - Change Management Controls
APP06 - Secure Extensions

This artifact combines all of the guidance into a cohesive and comprehensive document.

How To Use This Document

Both documents focus on the different aspects of securing a cloud ERP application. In the Top 20 Critical Controls for Cloud ERP Customers, a more general approach is provided, whereas in the Critical Controls Implementation for SAP, the working group has taken a more technical and granular approach.

Top 20 Critical Controls for Cloud ERP Customers

In the previous document, the working group elaborated on 20 critical controls that are required to secure cloud ERP applications. The following information is provided in that document:

- Domain: The domain assigned to the control
- Control Identification (ID): Unique name for the control
- Control Description: A description of the control and how it should be addressed
- Control Objectives: A description of what the control seeks to achieve
- Threats and Risks: Threats mitigated by the control, including those defined in the Treacherous 12: Top Threats to Cloud Computing 2016 report
- Related CCM Controls: If applicable, the IDs of the controls, as defined in the CSA CCM

Critical Controls Implementation for SAP

In this document, the working group focuses on providing guidelines on controls implementation as well as a set of checklists for SAP administrators. The controls implementation and the checklists apply to SAP NetWeaver(C) ABAP(C)-based Applications, and are generic enough to apply to all current versions, providing a detailed description of the control implementation, that can be complemented with external references that are also incorporated. The Control Implementation guidelines provide a detailed description of the control implementation and, combined with the Top 20 Critical Controls document previously released by the CSA, explains who would be typically responsible in an IaaS or SaaS scenario. However, please note that the actual responsibility for security depends on your contract with your supplier.

These checklists act as guidance only. The checklists provide general steps as well as some direction on how to carry out the implementation of the controls. The Checklist aims to be as technical as possible by providing SAP transaction numbers and other equivalent details. However, it is not feasible to provide that level of detail for a few controls. For example, BUS-03 (Continuous Compliance) is one such control. Instead, general guidance is provided. Lastly, specific references to SAP documentation are also provided.

Controls Implementation Part 1

Domain	Cloud ERP Application
Control ID	**APP01 - Secure Landscape**
Technology Stack	SAP NetWeaver ABAP
Versions	All
Control Implementation	The landscape in SAP NetWeaver Applications is composed of multiple SAP Systems that have diverse roles such as Sandbox, Development, QA or Production, to name a few. The security of the overall landscape is paramount to the security of the data that is hosted in the production system. In general terms, access to lower-risk systems such as development should not be potentially used to access a higher-risk system such as production. This means that as much as possible, all systems in the landscape should be separated (with different access controls) and secured with the same level and standards of protection.
Checklist	1. Make sure users in development are not authorized to access data in production without proper evaluation of their authorizations. As a general rule, assign authorizations that are as restrictive as possible across the landscape, independently of the system role (also known as the least privilege principle). 2. Make sure no RFC Destinations are configured from lower-risk systems to higher-risk systems, using stored credentials. For this, use transaction SM59, which can provide the login information. Exemptions may be connections to the TMS Controller and Solution Manager as long as they follow the least privilege principle. 3. Ensure S_RFCACL is properly assigned to users and restricted as much as possible in production environments. Use transaction SUIM to search for users and roles assigned with S_RFCACL.

1. Configure strong passwords for transport-related accounts such as TMSADM. Avoid using any default password for these accounts. Additionally, make sure the transport-related accounts are assigned only with the S_A.TMSADM profile. Use transaction SA38 and report RSUSR003 to identify the password of user TMSADM.
2. Ensure that right controls for insecure transport requests and insecure code are configured in the transport management system so it is not possible to move insecure objects into production.
3. Make sure the right approval process is set in the transport system, so all changes are properly approved by the right individuals before being moved across the landscape. Use transaction STMS to set up and validate the right approval process.
4. Ensure that any storage system used for the Transport Management System (Typically Common Transport Directory) is secured. If it is NFS or SMB based shares, these shares should be properly secured to avoid unauthorized access and modification of transport-related data.
5. Configure the "System Change Option" appropriately, according to the role each system fulfills in the transport process. This can be achieved globally (SE03/SE06) or per client (SCC4). Productive clients must be set to "not changeable".

References

- https://apps.support.sap.com/sap/support/knowledge/preview/en/1568362

Domain	Cloud ERP Application
Control ID	**APP02 - Baseline Secure Configurations**
Technology Stack	SAP Netweaver ABAP
Versions	All
Control Implementation	SAP Applications are complex as they are based on several components that interact with each other. These components are extensively configurable, and overall, SAP Systems are customizable and configurable. Approximately 10% of the configurations that can be modified and maintained have a security impact. The following components must be securely configured in SAP Applications: • SAP Application Server • SAP HTTP Interface • SAP Gateway • SAP Message Server • SAP Management Console
Checklist	1. Secure password policies configurations to match the corporate password policies. This can be achieved by maintaining the profile parameters through transaction RZ10 (maintaining both global and instance-specific profiles) or maintaining the user-defined security policies through transaction SECPOL. 2. Secure critical profile-parameters configurations using transaction RZ10 such as (but not limited to): a. No_automatic_user_sapstar = 1 b. RFC/callback_security_method = 3 c. Login/password_downwards_ compatibility = 0 3. Secure the diverse component ACL configurations such as: a. SAP Gateway sec info and reg info ACL files b. SAP Message Server ms acl info ACL file

4. Reduce the number of HTTP services that are enabled through transaction SICF.
5. Restrict access to SAP tables (SAP Applications can have up to 70,000 tables depending on its version and product). Be sure to:
   a. Configure an authorization group for all tables that don't have an authorization group assigned (Authorization Group unassigned or &NC&).
   b. Restrict S_TABU_DIS AND S_TABU_NAM authorization objects so these objects protect access to the most critical tables.
   c. Restrict SE11, SE16, SM30 and SM31 to address standard and custom tables.

References

- Secure Configuration of SAP NetWeaver Application Server Using ABAP
- SAP NetWeaver Application Server for ABAP Security Guide

Domain	Cloud ERP Application
Control ID	**APP03 - Security Vulnerabilities**
Technology Stack	SAP Netweaver ABAP
Versions	All
Control Implementation	On the second Tuesday of each month, SAP will release the security patches addressing security vulnerabilities that were either discovered internally by SAP or reported by external researchers. All of the patches must be evaluated, and a risk-based decision must be made, depending on the risk appetite of the organization as well as the potential business impact of each particular vulnerability. In addition, SAP guarantees that these security notes can be applied if the system is running on a Support Package Stack (SPS) not older than 18 months.
Checklist	1. Connect to security notes in the SAP launchpad: https://launchpad.support.sap.com/#/securitynotes 2. Get the list of SAP Security Notes released by SAP. 3. Categorize the "Vulnerability Trends Over Time" (i.e SAP: Vulnerability Statistics). 4. Identify the components affected by the SAP Security Notes as well as the SAP Systems that are affected by them. 5. Apply the relevant patches either through SNOTE or any other upgrade mechanism available to the technology stack (i.e. using the SPAM transaction). 6. Ensure that the SPS level is not older than 18 months (recommendation).
References	• SAP NetWeaver Application Server for ABAP Security Guide • CVE https://www.cvedetails.com/vendor/797/SAP.html

Domain	Integrations
Control ID	**INT01 – Secure Integrations and API**
Technology Stack	SAP Netweaver ABAP
Versions	All
Control Implementation	The extensive integration of ERP applications with outside applications and data sources is common practice because of the nature of processes supported by these systems. In a typical ERP environment, there are interfaces and connections between different solutions as well as different environments. If improperly secured, these integrations are ripe for abuse, and production information and data risks may be easily compromised. The management of interfaces across different ERP environments should address the following considerations: 1. Maintain an inventory of all interfaces, including the type of data that is exchanged and the technical details of the connections, such as protocol, user, business owner, authorizations and encryption details. 2. Avoid the use of insecurely provisioned interfaces, such as broad trust relationships or the utilization of usernames and passwords that others can leverage. 3. Always apply the "least privilege" principle to define the privileges that technical users will be granted for various interfaces. 4. If possible, encrypt all interfaces that exchange regulated or sensitive data between applications. 5. Avoid setting up interfaces from systems of lower security (such as development) to systems with higher security (such as production) whenever possible. 6. If secrets are used to set up the interfaces (i.e., API keys, passwords, certificates), establish the proper management process to govern those secrets (maintain/change/rotate if needed).

1. Ensure RFC Callback security in all systems, especially when systems from a higher risk classification connect to systems with a lower classification (e.g. Prod calls Dev)

For IaaS, PaaS—and possibly SaaS service models—this control is the security responsibility of the cloud customer.

Checklist

1. Define a unique identifier for integration and add it in your inventory of all integrations.
2. Use the principles of security by design and security by default. Design for mutual authentication between applications using client certificates, if possible.
3. Perform system hardening of public-facing components, including applications and infrastructure.
4. Create separate DMZ network segments, hosting securely configured SAP Web Dispatchers for publicly exposed integrations and API.
5. Incoming and Outgoing integration requests should be managed by a web application firewall or web proxy.
6. "Protect data-in-transit using protocols and strong crypto ciphers.
7. Enforce the principle of "least privilege" on the technical account used for the integration, to reduce consequences in case of a security breach.
8. Perform a pentest of the published integrations before business go-live, including initial vulnerability scan of the API. Run a basic network vulnerability scan, supporting CVSS rating, to measure your current situation. For publicly exposed systems, patch all vulnerabilities having CVSS score 4 and higher.

References

- SAP Process Integration Security Guide
- Security Information SAP Web Dispatcher
- CIS 20 Critical Security Controls

Domain	Cloud ERP Data
Control ID	**DAT01 – Continuous Monitoring**
Technology Stack	SAP Netweaver ABAP
Versions	All
Control Implementation	SAP Applications are complex and built on top of multiple components. To understand what is happening within an SAP Application, multiple sources of data must be enabled and analyzed. This has to be driven by a continuous monitoring program which includes an incident response program with the following components: 1. Enable the logs and traces that are relevant for SAP Applications (such as Security Audit Log and HTTP access log), sending them to a centralized log and security server. 2. Implement a process to review the logs periodically, preferably using a SIEM tool, so a timely response is possible. 3. Implement an incident response process so whenever an incident is identified across SAP applications, the proper teams are involved to contain the incident. In diverse versions of SAP Applications, even Cloud multitenant, Read Access Logging (RAL) can be used to log who had accessed sensitive data; this access logging is enabled by the company per their definition of sensitive data.
Checklist	Determine which data must be logged under which circumstances. The organization must define which legal, compliance or security requirements to apply and which data must be logged. 1. Ensure that the following logs are enabled in SAP applications by incorporating this in the baseline configuration of every new instance of SAP: • Security Audit Log (through transaction SM19) • SAP Gateway Log (through transaction SMGW)

- SAP Table Change logging (by enabling parameter rec/client and transaction SE13)
- HTTP access log (SMICM)
- Message server log
- Change documents
- Read Access Log

NOTE: Enable security-relevant events that are meaningful to your organization and keep in mind that enabling all might pose a performance and storage impact. Additionally, it is important to understand which data must be logged under which circumstances (e.g. salary information, Social Security number, or bank account)

2. Implement a process to review the logs periodically by analyzing the generated events against a list of previously defined potentially insecure behaviors. The generated logs can be accessed using the following transactions:

    - Security Audit Log (through transaction SM20)
    - SAP Gateway Log (through transaction SMGW)
    - SAP Table Change logging (through transaction SCU3)
    - HTTP access log
    - Message server log
    - Change documents

Implement a process to escalate and contain incidents in SAP Applications. This might involve actions such as:

- Locking a user account (Transaction SU01)
- Changing users' passwords (Transaction SU10)
- Further reviewing access logs and any other source of information from a consolidated point of view

References

- SAP Audit and Logging
- The SAP Security Audit Log
- Activate/Deactivate Table Change Logging
- Performance Problems through Table Logging
- Performance: Log Table DBTABLOG Increases in Size Due to KONP
- Read Access Logging

Domain	Cloud ERP Data
Control ID	**DAT02 – Data Separation**
Technology Stack	SAP Netweaver ABAP
Versions	All
Control Implementation	Business data is critically important in ERP applications. This data is typically stored in a database and provides access to multiple users and application servers. Additionally, in a typical ERP landscape, there are numerous environments (i.e., development, quality assurance, and production), as well as tenants. Data must be segregated appropriately in ERP systems and environments. In other words, production data should not be available in non-production environments, and any data segregation should be executed appropriately at the application level (i.e., concepts of systems, clients, tenants or company codes). Consider the following during the implementation of the ERP application: 1. Build the landscape in layers, with firewall separation of the production system, testing system and the development systems. 2. Separate production data from non-production data and avoid copying production data from production environments without proper sanitization. 3. Properly configure and implement any client or tenant separation—particularly when the cloud customer configures it—so that no user has access to both production and non-production tenant. Regardless of the service model, this control is the responsibility of the cloud customer.

Checklist

1. Build a landscape with clear separation of development, test, quality assurance (optional) and production to implement controlled change management. Use transaction STMS to develop and check the proper set-up of all the transport mechanisms.
2. Implement logical separation of production and non-production network.
3. Consider subzoning of production area depending on information trust domains.
4. Implement a management network with jumphosts and patch servers for privileged administration. Multi-factor authentication should be implemented.
5. Implement a DMZ for external access. Consider having an inner DMZ even for internal access depending on the threat model.
6. Monitor infrastructure and applications using a SIEM solution

References

- SAP NetWeaver Security Guide
- SAP NetWeaver Security Guide 7.5
- Using Multiple Network Zones
- SAP Cloud Platform Connectivity

Domain	Cloud ERP Data
Control ID	**DAT03 – Data Encryption**
Technology Stack	SAP Netweaver ABAP
Versions	SAP NetWeaver All Versions CommonCryptoLib v8.5
Control Implementation	Business data stored and processed by the ERP application is its most crucial component. Sensitive data at rest must be encrypted and classified to avoid unauthorized access according to predefined rules and policies. Avoid encrypting business data with the same key. The organization must first define data governance policies, such aswhat data should be encrypted and what should not (i.e., encryptionof all business data could render the ERP application useless). Concerning ERP data, adhere to the following guidelines: 1. Data should be encrypted while at rest and when stored in the database or any other location. 2. Encrypt data during transmission to the end-user. If the interface is web-based (as it is for the majority of ERP applications), then make sure to implement transport-level encryption with robust protocols and ciphers. 3. If using encryption keys and certificates, ensure the proper process is in place to maintain, issue, revoke and control access to these keys and certificates. 4. Database encryption will not protect you from injection attacks but may impact cost and performance, thus it should be thoroughly analyzed. 5. Offline ERP data should be encrypted and password protected using a standard protocol such as AES-256 (or better) having a password matching the strength, i.e. 24 characters random generated password. For IaaS, this control is the responsibility of the cloud customer. For PaaS and SaaS, the customer must conduct due diligence and ensure the cloud provider is adequately protecting their data.

Checklist

1. Secure data-at-rest on server-server by enabling full disk encryption in the operating system.
2. Secure data-at-rest on the client-side by enabling full disk encryption in the operating system.
3. If using SAP HANA Database, leverage standard encryption mechanisms for all the information in the data area and log volume area using SAP HANA administrator.
4. Secure data-in-transit by enabling SNC protocol for SAP GUI applications, using SSO and configure for protocol Kerberos and AES-128.
5. Secure data-in-transit by enabling secure communication in SAP Web Dispatcher using strong protocols and crypto ciphers, i.e TLS1.2 with AES-128.
6. Run a basic network vulnerability scan, supporting CVSS rating, to measure your current situation. For publicly exposed systems, patch all vulnerabilities having CVSS score 4 and higher.
7. Authorize a file compression software supporting AES-256 encryption to protect offline data.

References

- SAP NetWeaver Security Guide
- Setting up SSL on Application Server ABAP
- CommonCryptoLib 8 cryptographic algorithms
- Security Information SAP Web Dispatcher
- SAP HANA Encryption
- Ecrypt CSA - Algorithms, Key Size and Protocols Report, 2018
- Commercial National Security Algorithm Suite and Quantum Computing FAQ

Domain	Business Processes
Control ID	**BUS01 - Inventory of Business Assets, Data and Processes**
Technology Stack	SAP Netweaver ABAP
Versions	All
Control Implementation	Business data stored and processed by the ERP application is its One of the most challenging parts of operating business applications at scale is to have the right level of visibility around the data and the processes that each application is supporting. Having a clear inventory of these components is the starting point to understand where the crown jewels are in an organization, and to be able to provide the right governance and controls around those components. Implement an inventory of applications, data and processes which serve as a single source of truth in regard to business processes.
Checklist	1. Before starting, the business should provide or walk through the 'business process flow'. Risks should be identified in the process flow to which controls are applied. For all automated controls in the process flow, an asset is identified that supports/provides the control point. This Asset list when compiled is critical to the business as it documents the "assets" critical to the survival of the business. 2. Identify all the technical components and SAP Applications that build up the SAP environment. Incorporate non-SAP Applications if these are also critical components of the business processes running through SAP. 3. If the company is using a single source of truth (inventory/repository) for SAP applications, make sure it is properly maintained and up to date. If there are many inventories/repositories, identify all and create a process to condense all the information. Some options for these repositories are SLD and LMDB.

1. With the functional leads, identify the key business processes that are supported by each SAP Application and document. This should be a key input to identify the overall criticality of an SAP System.
2. Check the process for creating new SAP Systems as well as all existing environments to validate if these were purposely created and if the right approvals were in place.
3. Validate that the right stakeholders (IT, BASIS, Information Security) are aware of the service agreements regarding updates of security configurations, software components and patches across SAP Applications.
4. Implement software components and patch management process that can provide visibility on missing patches and outdated software components.
5. Check the Software components that are installed on each SAP System (System-->Status) and keep an inventory of all systems and business processes, capturing the importance to the business of each technical asset.

References

- SAP System Landscape Directory
- Landscape Management Database

Domain	Business Processes
Control ID	**BUS02 - Business Process Controls**
Technology Stack	SAP Netweaver ABAP
Versions	All
Control Implementation	SAP Applications support several critical business processes. Controls must be put in place to ensure that no fraudulent activities can be executed by abusing existing or elevated privileges.
Checklist	1. Together with the business process owners, identify critical steps on each one of the critical business processes. 2. Identify the systems involved, particularly the SAP Applications that are supporting these processes, including the systems running on the cloud. 3. Identify system interfaces and define the purpose of each of them. Filter and/or disable those systems' interfaces that are not required: a. Delete RFC Destinations (Transaction SM59) b. Enable UCON (Unified Connectivity, transaction UCONCOCKPIT) c. Disable ICF Services (Transaction SICF) d. Filter system ports 4. Develop specific business controls for business steps that require moving data among environments a. Extra authorizations/privileges (Through PFCG roles modification) b. Approvals from managers 5. Implement a process that involves automatic processes controls as well as the monitor for the usage of interfaces related to the business processes.
References	• End to End Business Processes in SAP

Domain	Business Processes
Control ID	**BUS03 - Continuous Compliance**
Technology Stack	SAP Netweaver ABAP
Versions	All
Control Implementation	Due to the nature of data and processes that SAP Applications support, it is key to maintain certain levels of compliance with the regulations that are applicable to the data, processes and industry that the organization is operating in. Implement a process that ensures continuous compliance and can work as a centralized view to monitor control effectiveness in real time.
Checklist	1. Identify compliance and regulatory standards that are affecting the SAP Applications. 2. Identify the specific key controls that must be in place. 3. Identify the required testing procedures to validate the operating effectiveness of those controls. 4. Develop automated testing procedures that can validate controls effectiveness 24x7. 5. Implement an alerting mechanism to address audit findings as soon as they happen.
References	• https://en.wikipedia.org/wiki/Continuous_monitoring • https://en.wikipedia.org/wiki/Continuous_auditing • https://en.wikipedia.org/wiki/Regulatory_compliance

Controls Implementation PART 2

Domain	Cloud ERP Users
Control ID	**USR01 - Secure Authentication**
Technology Stack	SAP Netweaver ABAP
Versions	All versions
Control Implementation	The authentication mechanism enabled for the users of the SAP Application should be configured in a secure way so no one can impersonate application users. This calls for single sign-on, strong password policies, and additional factors during the authentication as well as using a secure protocol for the authentication process.
Checklist	The following attributes should be true for the authentication process: 1. The communication protocol is encrypted in a way that no man-in-the-middle attacks are possible. To ensure an encrypted authentication process, SNC should be used to encrypt the SAPGUI authentication and TLS to encrypt the Web-based (HTTPS) authentication processes. 2. This should ensure the communication protocol is secure so no replay attacks are possible. 3. Strong password policies should be enforced for users accessing the Cloud ERP Application. This should be achieved by enabling strong policies through the password policy profile parameters (by system/application server) or by using transaction SECPOL (by user). 4. Improved Security: a. If possible, the authentication process should ask for more than one factor (i.e. the password and a time-based token). This is especially important for high-privileged users such as the SAP BASIS team members.

b. If possible, single sign-on schemes should be enabled so the user doesn't have to remember a specific password (or set of passwords) for each ERP Application and for each SAP Client. Since there will always be a subset of user accounts that can bypass single sign-on, checklist item #2 is still important in an SSO environment.

References

- SAP NetWeaver Security: Authentication and Single Sign-On
- User Authentication and Single Sign On
- Secure Network Communications (SNC)

Domain	Cloud ERP Users
Control ID	**USR02 - User Accounts Management**
Technology Stack	SAP Netweaver ABAP
Versions	All versions
Control Implementation	Managing the User Accounts on SAP Applications is paramount for ensuring only the appropriate people have access to the system. In the SAP Netweaver-based family of applications, the user base is extended as the concept of client (MANDT) broadens the potential for access to the system. This control aims to provide assurance that the user accounts that exist on the SAP Application are properly controlled and governed to avoid unauthorized access to the business information.
Checklist	1. **Default Users:** Appropriate agent/entity must ensure that no default users (SAP*, DDIC, TMSADM, EARLYWATCH, etc...) are configured with default passwords on any SAP Client, including the standard SAP Clients. To do this, transaction RSUSR003 should be executed. 2. **Technical Users:** Agent/entity must ensure that there is a process for creation of technical users and that those technical users are created using specialized roles and not generic roles or profiles such as SAP_ALL. For each technical user, transaction SU01 must be executed, accessing the profiles tab to validate that the assigned profiles are related to specialized integration or business roles. 3. **Default Clients:** Special attention should be given to non-productive clients to ensure that whatever user provisioning process is in place, it applies to those clients too (000, 001 and 066 whenever applicable). Appropriate agent/entity should connect to each SAP Client and review the users created in those clients to make sure each user has a valid business purpose, an effective company employee associated to it and that it was created according to the security recommendations (i.e. strong password, minimum authorizations).

References

- Standard Users in SAP Netweaver JAVA
- SAP Netweaver ABAP - Protecting Special Users
- Standard Users in SAP HANA
- SAP Netweaver ABAP - Protecting Standard Users

Domain	Cloud ERP Users
Control ID	**USR03 - Role-based Access Control**
Technology Stack	SAP Netweaver ABAP
Versions	All versions
Control Implementation	Users in the same functional or technical area have the same authorizations bundled together in profiles which are assigned to single roles or composite roles. Roles can be combined in composite roles. Single roles or composite roles are assigned to the user which gives the user the permission. There are three major areas that need to be monitored and audited constantly to ensure that users have only the permission they should get and use for their day-to-day business: The most important topic for the control implementation is to screen the change management process of user, authorization, roles, and profiles, e.g. when an employee changes position and the employee's authorization needs to be changed too. A process should be established when an HR change to a position happens. A new hire as well as a termination should be monitored to take appropriate action.
Checklist	Check the following topics regularly: 1. Appropriate entity should use transaction SUIM to track user changes and to check if the changes are according to the line of business and the task given to the employee. 2. New hires, terminations and position changes should be monitored constantly. New hires (via FLUCTUATIONS query) must have been logged in within a certain period, and their password. Terminations (via FLUCTUATIONS query) should be locked. In addition, their authorizations, roles and profiles should be removed from their user master record (transaction SU01).

3. Appropriate entity should use transaction SUIM to track changes to authorization, roles and profiles and how those are reflected in the overall authorization concept, also checking if there are changes made by users who are not entitled to make changes (e.g. outside authorization team). Transaction SUIM should be used. It should be noted that organizations that make changes in a development environment and then test and migrate to production will have these change documents in the development system, not the productive system.
4. Appropriate entity should use transaction RSPFPAR to check relevant profile parameters:
   - auth/check/calltransaction = 3
   - auth/new_buffering = 4
   - auth/no_check_in_some_cases = Y
   - auth/object_disabling_active = N
   - auth/rfc_authority_check = 8
   - auth/tcodes_not_checked = empty or SU53 or SU56
5. It is necessary to regularly check users who have a security policy different from that applied to the complete system. Appropriate entity should use transaction SECPOL and check which users have lower security settings than others.

Note: If SECPOL is being used, then every policy should have a value for every parameter enabled by SECPOL, otherwise these parameters will take SAP's default values.

6. Switchable Authorizations: Once specific further authorization checks through transaction SACF have been switched on, they must not be switched off again.

References

- Role-based Access Control (RBAC)
- NIST RBAC model
- SAP help Authorization Concept
- organization-based Access Control (OrBAC)
- SAP Help (structural authorization)
- context-sensitive Access Control (SAP Help)
- Switchable Authorization (SACF)
- SECPOL for User-based Password Policies

Domain	Cloud ERP Users
Control ID	**USR04 - Emergency Access**
Technology Stack	SAP Netweaver ABAP
Versions	All versions
Control Implementation	Emergency Access is granted to staff upon system failures, errors or, in the normal course of business, the unavailability of personnel. Granting emergency access results in privileges normally prohibited, exposing the entity to additional risk which must be properly managed. 1. Determine Emergency Access policy - who can do what, when, where and how. Address responsibilities for administration, granting, monitoring, termination and reporting and auditing of emergency access. Configure according to policy: 2. Configure Roles - who can approve and or do what 3. Grant Access - who can approve and for what activities 4. Monitor Access - what gets reported, when, to whom, and how 5. Terminate Access - who can approve 6. Report and Audit Access - what, when, to whom, and how
Checklist	Configuring the roles according to policy: 1. Administrators have total access except to logs (read only access). They need the ability to assign Firefighter IDs to business process owners and to firefighters. They also need to be able to run reports, maintain data tables, and make sure reason codes and the table are current. Administrators should be able to enable email notifications for controllers through the firefighter assignment function and through Customizing. Standard Role: SAP_GRAC_SUPER_USER_MGMT_ADMIN 2. Firefighters need the ability to request emergency access using a self-service request. The request must include access

to the required activities. The request should be for a period of time.

Standard Role: SAP_GRAC_SUPER_USER_MGMT_USER

3. Business process owners need the ability to review and approve or reject requests as well as the possibility to review firefighter activity via EAM reports and logs (read-only to all).

   Standard Role: SAP_GRAC_SUPER_USER_MGMT_OWNER

4. Controllers (compliance owners) need the ability to perform periodic activity audits as well sign-off via the EAM reports and logs (read-only to all). This access is achieved through the Standard Role SAP_GRAC_SUPER_USER_MGMT_CNTLR

References

- CSA Top 20 ERP Controls USR04 - Emergency Access
- SAP Using Emergency Access Management
- SAP Maintaining Configuration Settings in Access Control
- SAP Security Guide for SAP Access Control, SAP Process Control and SAP Risk Management

Domain	Cloud ERP Users
Control ID	**USR05 - Segregation of Duties**
Technology Stack	SAP Netweaver ABAP
Versions	All versions
Control Implementation	Segregation of duties (SOD) is a control principle which supports the idea that no one should control a process from beginning to end and that within any process no one should have the ability to perform more than one incompatible task such as transaction approval, accounting and reconciliation. A combination of these tasks could, for example, allow someone to write off accounts receivable or in the IT area make unreviewed changes to programs. SOD entails walking a line between task and employee flexibility and security. SAP access is created using a role or task-based method. If the environment is static and responsibilities within roles do not change much, then role-based (what an AP person does, for example) access might be better. If the environment is dynamic and responsibilities within roles change frequently, then a task-based (multiple roles each representing a specific task) design might be more advantageous. Roles are based on the organizational plan of the company and are the connection between the user and authorizations. Users are assigned roles which contain the authorizations with which users can access transactions, reports, Web-based applications, etc. When users log on, they are presented with menus that display their access.
Checklist	1. Define organization and hierarchy. 2. Define and maintain user and administrative roles and profiles using transaction PFCG - Role Maintenance. a. Generating users' roles from the standard library using the Profile Generator, which automatically creates authorization data based on selected menu functions, can

save substantial time and effort. These roles can then be customized. SAP recommends using the role maintenance functions and the profile generator (transaction code PFCG) to maintain the roles, authorizations, and profiles.

   b. Be sure to restrict access to critical transactions, programs, remote function calls, database tables, web services, etc.

3. Make sure users do not have technical administrative profiles (done in SU01) such as SAP_ALL.
4. Create a SOD matrix using the business process rules and best practices that described above. To generate a pleasing visual SAP GRC or other commercial software needs to be used.
   a. Alternatively, the SAP User Information System (SUIM) transaction can be used to generate various reports to analyze users, roles, profiles, authorizations, authorization objects, transactions, comparison, where-used lists and change documents.
   b. Also, RSUSR008_009_NEW can be used with critical authorizations to perform an analysis of users with critical combinations of authorizations. Before this report can be run, critical authorizations must be loaded. This can be done via an Excel sheet.
5. In any case, conflicts must be risk-assessed. Low-level risks might be ignored. Medium risks will need a closer review. High-risk SOD conflicts need to be resolved either by
   f. separating conflicting roles, or if there is a problem within a role, altering the role. If a role needs to be altered, it is recommended that a copy be made and changed; or
   g. through alternative controls such as review, and approval of all activities performed using the conflicted duties on whatever periodic basis is appropriate.
5. After each conflict or conflicts are resolved, a new analysis should be run to verify that conflicts have indeed been resolved.

References

- CSA Top 20 ERP Controls USR04 - Emergency Access
- SAP Segregation of Duties
- Wagener, M. (2008). A Practical Guide for SAP Security
- SAP Security Guide for SAP Access Control, SAP Process Control and SAP Risk Management

Domain	Cloud ERP Users
Control ID	**USR06 - Secure User Provisioning/Deprovisioning**
Technology Stack	SAP Netweaver ABAP
Versions	All versions
Control Implementation	Provisioning technical and functional users within an SAP Application is a key security control from an operational perspective, as it needs to ensure: 1. there is a valid business reason and requirement behind the creation of the user. 2. the user is created in the appropriate system and client (tenant). 3. there is a responsible person associated with this account. 4. the permissions are assigned according to the least privilege approach. 5. authentication and password settings are provisioned accordingly and securely; and 6. There is a validity period defined whenever applicable to the account. Unmanaged and dormant accounts could be misused to access the system in an unauthorized way; therefore, the de-provisioning of user's accounts should be properly managed. It is recommended to use a centralized system for managing user accounts across the different SAP Systems and landscapes. An Identity Management system would ensure that when a user is provisioned it is done across all systems and that when a user is deprovisioned, that happens across all systems as well.
Checklist	1. Ensure the provisioning and deprovisioning of users is properly managed and documented in a company policy. 2. Check users' last login through the security audit log, which requires enabling rsau/enable=1, turning on the "successful login" event in SM19, and then monitoring in SM20.

3. Lock or expire users that are identified as dormant or non-active using transaction SU10. Best practice is to expire the ID with a validity date, remove all roles and profiles, and put the user in a restricted user group (i.e. TERMINATED).

References

- Integrating a Central User Administration System
- 2072086 - CUA User Provisioning from GRC to CUA Parent/Child
- User Provisioning

Domain	Cloud ERP Users
Control ID	**USR07 – ERP Accounts Security**
Technology Stack	SAP Netweaver ABAP
Versions	All versions
Control Implementation	Security of ERP accounts is paramount, especially as ERP applications migrate to the cloud. Login processes should be developed to make it more difficult for adversaries to successfully enter the ERP system using valid compromised credentials. Additionally, the ERP application should implement session management mechanisms according to the applicable standards. Organizations using cloud ERP applications should be sure to: 1. enable multi-factor authentication. 2. enable the passwords rule, prioritizing password length rather than complexity. 3. analyze user login behavior to detect unfamiliar login location, time, etc. Also, link to other user activities (e.g., data provided through cloud access and security broker providers, as well as conditional access). 4. implement single sign-on property. 5. limit access to the ERP system from specific networks. 6. ensure session tokens are dynamic, sufficiently random, encrypted and expire on time; and 7. enable audit logging of the user activities and transactions (read-only to all, including administrators). In IaaS, this control is the responsibility of the customer. In PaaS and SaaS, the management of this control is sharable. For example, multi-factor authentication may be available through the CSP, but the customer must activate it.
Checklist	1. Enable encryption between client and server to protected user credentials. 2. Enable multi-factor authentication for users.

3. Where multi-factor authentication is not supported, accounts should use passwords that are unique to that system.
4. Enable client certificate authentication for applications.
5. Limit access to ERP systems by network segmentation between clients and servers.
6. Limit access to ERP from specific networks. Consider limiting access to ERP system by implementing software defined perimeter.
7. Use wired networks for high security tasks and critical clients, such as IT operations management networks or backoffice.
8. Enable security audit logging (set read-only to all)
9. Send the audit logs to a centralized log server (server is not accessible by SAP administrators and users).
10. Monitor the centralized log server using a SIEM product, using a SAP aware product.

References

- CSA Software Defined Perimeter for IaaS
- CIS 20 Critical Security Controls
- NCSC Cloud Security Guidance

Domain	Cloud ERP Apllication
Control ID	**APP04 - Secure Communications**
Technology Stack	SAP Netweaver ABAP
Versions	All versions
Control Implementation	In an SAP environment there are multiple ways to communicate with the system. It can be either a system-to-system communication or an interaction between a user UI and the SAP system. The user UI might be a Web-Frontend, SAPGui, Excel or other front-end application like Eclipse. It is very important that the communication stream is encrypted to not allow sniffing passwords or business data. The following are the most important areas to highlight when it comes to securing communications: 1. SAPGui Security (SNC for DIAG) 2. Web Application Security (HTTP(s)/SSL), e.g. for ICF 3. Office Front-end Integration Communication (RFC SDK) and other SAP or non-SAP (3rd Party) Application like SAP TMS or Eclipse (SNC for RFC)
Checklist	1. Limit the attack surface 2. UCON, limit RFC communication 3. SAP Gateway, who can talk to the SAP system from outside (internally and externally) 4. ICF services enable only these services being used 5. Secure the communication with appropriate authorizations. The most important authorizations include: a. S_RFC - Auth Check for RFC Access b. S_RFC_ADM - Administration for RFC Destinations c. S_RFCACL - Auth Check to use RFC Search Help d. S_RFCACL - Auth Check for RFC User (trusted RFC)

e. S_ICF - Auth Check for ICF Access
f. S_ICF_ADM - Administration for ICM / ICF
g. Finally, functional authorizations should be implemented in all the RFC function modules or WebServices that are called from outside an SAP system

9. Secure the data stream of the interfaces / communication itself, using Secure Network Communication (SNC),
   a. enable SNC for SAPGui and do regular administration tasks: check System Profile Parameter snc/enable = 1 (transaction RSPFPAR)
   b. check System Profile Parameter snc/accept_insecure_gui = U (transaction RSPFPAR)
   c. SNC for RFC communication / connections is the second common scenario: check System Profile Parameter snc/accept_insecure_rfc = U (transaction RSPFPAR)
11. WebServices and HTTP-based protocols also need protection. This is accomplished via the Transport Layer Security Protocol (TLS).
12. RFC Call back
   a. RFC Call Back is one of the most known and most common attack vectors for internal and external SAP RFC connections.
   b. Customers can protect themselves generally by setting the SAP System Profile Parameter rfc/callback_security_method = 3 (transaction RSPFPAR)
   c. Callbacks can be checked if using transaction SM59. If it does not show a green light "RFC callback check secure" the respective SAP system is not protected.

References

- Unified Connectivity (UCON)
- SAP Gateway Security
- Security Measures Overview (ICF)
- Secure Network Communications (SNC)
- Use of network security products with SNC
- Steps to enable and configure SSL
- J2EE Engine - How to configure SSL
- Usage of SNC in SAPs transport layer (TMS): HowTo.
- Configuring the use of SAPCRYPTOLIB for SNC
- Note 1848999 Central Note for CommonCryptoLib 8 (SAPCRYPTOLIB)
- 2338952 – CommonCryptoLib 8.5: Configuration Profile Parameters
- Algorithms, Key Size and Protocols Report (2018), ECRYPT CSA

Domain	Cloud ERP Apllication
Control ID	**APP05 - Change Management Controls**
Technology Stack	SAP Netweaver ABAP
Versions	All versions
Control Implementation	The appropriate change management controls should be implemented across the SAP Landscape so no unmanaged changes can be implemented in the production system through unauthorized access. Several mechanisms enforce change management across the SAP Applications Landscape: 1. SAP Transport Management System 2. SAP CTS/CTS+/Charm 3. Closing the SAP System for changes 4. Enabling the QA Approval Procedure for change management control It is important to emphasize that for SAP Applications, most changes can be transported through the transport system; therefore, it is crucial to place the right control process for changes flowing into production A process designed to detect unauthorized or malicious changes should be placed as a minimum in the QA system, with validations in the development system as well.
Checklist	1. Use transaction SCC4 to control the Client-level Change Options (depending on the company policy). 2. Use Transaction SE06 to control the System-level Change Option (depending on the company policy). 3. In transaction STMS (Client 000 in the Domain Controller for the SAP Domain), ensure that the QA approval procedure is configured with the proper approval methods, as defined in the company policy.

4. Ensure authorizations that control changes through the landscape are properly set. Some examples of these authorizations are S_TRANSPRT and S_CTS_ADMI.
5. Ensure that only a limited and controlled number of users have access to SAP Standard Clients 000 and 001. The same principle should apply to users with access to the customizing related transactions (i.e. SPRO, SPRO_ADMIN).

References

- Change Management in the SAP System Landscape
- Setting Up User and Authorization Administrators
- Authorizations in the CTS

Domain	Cloud ERP Apllication
Control ID	**APP06 - Secure Extensions**
Technology Stack	SAP Netweaver ABAP
Versions	All versions
Control Implementation	The way SAP allows for customizing the business processes is called extensions. These can be done in-app or side-by-side. During their build-phase quality and security must be considered for all stacks like authentication, authorization, communication security and finally code security. For further checks e.g. OWASP Top 10 from the OWASP foundation these days are a given. These checks are mainly for Web Applications and might not fit for other programming areas/ languages which are proprietary in SAP.
Checklist	No changes shall be made to SAP standard development objects unless absolutely required. The Enhancement Framework offers a better alternative to the modification approach. It enables you to add functionality to standard SAP software without actually changing the original repository objects and to organize these enhancements separately from the enhanced objects. Enhancement type to use should be chosen in the following order: 1. Find and use a Business Add-Ins (BADI) 2. Try solve it using Explicit Source Code, Function and Class Enhancements 3. Traditional Customer Exit (Function) 4. Business Transaction Events 5. Traditional User Exit (Form) 6. Implicit Source Code Enhancement, only when there is no other available alternative

Modification of SAP standard objects outside of the above should not be permitted unless explicitly recommended by SAP in writing.

OWASP TOP 10 must be considered for any ABAP development. The application should never trust any input from the user or client. In the end, it comes down to performing a security and compliance check for any new code that is going to be running in the SAP application or integrated into it as an extension. Some examples of what to check for:

7. Check all OWASP top 10
8. Check all BIZE TEC/11, APP/11 and HANA/11
9. Security Guidelines that are best practice e.g., SAP Development Guideline from DSAG
10. Those checks should all be included in a company policy that should be followed by developers.
11. A company policy is usually the first step towards code quality and code security to allow "externals" to check against the policies (auditors, internal InfoSec Team, internal compliance team, etc.) but also to have KPIs for management and development

What to organize and establish (process):

1. Make security and quality part of the complete development process from the very first start (architecture, design) to the very end (Unit Tests, atomized code scans).
2. Make usage of scan tools mandatory.
3. Supplement these checks with peer to peer review for code to transfer knowledge.
4. Integrate in IDE of developer's choice to make code scan part of the daily routine of a developer and to give instant feedback about what must be changed and what is not a good security measure.
5. Establish process so that technical stop / WF is implemented when code does not comply with policies for code quality and code security.

References

- DSAG Development Guide for ABAP.
- The VirtualForge (Onapsis) Benchmark gives approximately 1 severe security defect in 1000 Lines of Code in the area of Security, Compliance and Data leak Prevention.
- Enhancement Framework

- OWASP TOP 10 vs ABAP developer
- ABAP Security Notes
- Secure Programming - ABAP
- OWASP Top Ten Web Application Security Risks | OWASP