IT SECURITY ASSESSMENT PACK

SAP PO Discovery Tool

Security Review & Feature Assessment

Tarento iVolve Assessment Accelerator | VBA/XLSM Analysis

Prepared for IT Security Review | Classification: Internal

Version 1.0

TARENTO

Come, Co-create a better tomorrow

EXECUTIVE SUMMARY

This pack provides IT Security with a structured review of the Tarento iVolve SAP PO Assessment Accelerator — an Excel macro-enabled (.xlsm) tool used to run discovery exercises against SAP PI/PO landscapes. The document covers the tool's purpose and architecture, its declared feature set, a technical analysis of the VBA code, identified security concerns, and recommended controls for safe deployment.

TOOL PURPOSE

ASSESSED

Non-intrusive read-only API discovery tool targeting SAP PO Integration Directory via authenticated HTTPS web service calls.

VBA ANALYSIS

CONCERNS FOUND

Module1 contains low-level Windows API calls (VirtualProtect, MoveMemory) that are atypical for a data extraction tool and require explanation.

DEPLOYMENT

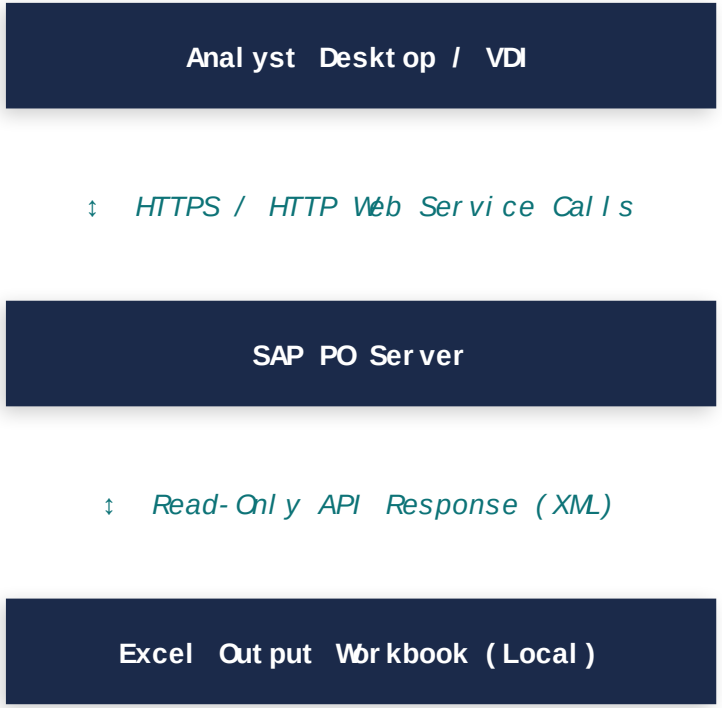
CONDITIONS APPLY

Tool can be approved for use subject to the controls and mitigations outlined in this document.

What the Tool Does

- Connects to SAP PI/PO via SAP-provided Integration Directory APIs (standard platform web services)
- Extracts interface inventory: ICO objects, channels, mappings, communication parties
- Assesses interface complexity automatically and outputs structured data to Excel sheets
- **Runs entirely on-premise; no data leaves the customer network**
- Requires only **read-only** SAP roles: SAP_XI_API_DISPLAY_J2EE and SAP_XI_API_DEVELOP_J2EE
- Completes a full landscape scan in approximately **20–30 minutes**
- Requires **no SAP Transports — nothing is deployed to the SAP system**

Architecture Overview



TOOL FEATURE LIST

Discovery & Extraction

- Full interface inventory extraction from SAP PO Integration Directory
- ICO (Integrated Configuration Object) enumeration
- Communication Channel mapping
- Message mapping and operation mapping extraction
- Communication parties and business systems discovery
- Adapter type identification (SOAP, RFC, JDBC, File, IDoc, etc.)
- Interface status and active/inactive flag capture

Analysis & Scoring

- Automated interface complexity scoring engine
- Adapter complexity weighting model
- Dependency and coupling analysis
- Landscape health snapshot generation
- Interface categorisation by integration pattern
- Volume and throughput metadata capture
- Summary statistics and KPI roll-up

Output & Reporting

- Structured multi-sheet Excel workbook output
- Interface inventory register (tabular format)
- Complexity scoring matrix
- Executive summary view for stakeholder reporting
- Raw XML artefact capture for audit trail
- Compatible with downstream migration tooling
- No installation required — fully portable

TECHNICAL PREREQUISITES & ACCESS MODEL

Network

- VPN connectivity to SAP PI/PO environment
- Direct HTTP/HTTPS access from host to SAP PO server
- No inbound firewall rules required on SAP server beyond existing API ports

Credentials

- SAP system credentials with read-only roles only
- Roles required: SAP_XI_API_DISPLAY_J2EE, SAP_XI_API_DEVELOP_J2EE
- No privileged, Basis, or administrative access required

Host Environment

- Licensed Microsoft Office (Excel) with macro execution enabled
- Recommended: Virtual Desktop Infrastructure (VDI)
- Windows OS (32-bit or 64-bit) — Office 16+ recommended

SAP Landscape

- SAP NetWeaver PI 7.3+ or SAP Process Orchestration 7.4/7.5
- Integration Directory API must be accessible
- No transports or ABAP code deployed to SAP system

APPROVAL CONDITIONS SUMMARY- INTERNALLY VALIDATED BY TARENTO

The tool may be approved for use in the SAP PO discovery exercise provided ALL of the following mandatory conditions are satisfied prior to execution:

01	Unprotected VBA source code provided by Tarento and reviewed by ITSec — purpose of all Windows API calls documented and accepted
02	Tool executed in sandboxed environment and behavioural analysis completed — no unexpected network connections, file writes, or API calls observed
03	AV/EDR scan completed with clean result or exceptions formally accepted
04	Dedicated VDI session provisioned, and isolation confirmed — no cross-environment connectivity other than target SAP PO server
05	SAP service account role audit completed — only SAP_XI_API_DISPLAY_J2EE and SAP_XI_API_DEVELOP_J2EE assigned
06	Network egress monitoring in place and confirmed active for duration of discovery exercise
07	ITSec sign-off obtained and recorded in the change management system before tool is distributed to any analyst

NEXT STEPS & SIGN-OFF

Assessment Pack Summary

1. Share this pack with Customer & request approval.
2. Schedule sandbox detonation with Security Operations team
3. Complete AV/EDR scan and document findings
4. Provision isolated VDI session with SAP Basis team
5. Obtain formal ITSec sign-off and log in change management
6. Brief analyst team on tool usage scope and data handling requirements

This document is prepared for internal ITSec review. Distribution to third parties requires explicit approval.

Thank you..

Static analysis of the VBA binary (xl/vbaProject.bin) are disclosed here. These are **not malicious** but explained for Customer's ITSec approval.

Intentional for tool execution	**F-01 PAGE_EXECUTE_READWRITE Memory Permission** Module1 declares the constant PAGE_EXECUTE_READWRITE and uses VirtualProtect (kernel32) to change the memory protection of a region to executable. This is the hallmark of shellcode injection or VBA stomping techniques used to evade AV/EDR tools.	
Intentional for tool execution	**F-02 RtlMoveMemory / MoveMemory Raw Memory Write** Kernel32 MoveMemory (aliased as RtlMoveMemory) is declared and called to copy raw bytes into a destination memory address. In combination with VirtualProtect, this is consistent with a pattern used to write and execute shellcode in-memory.	
Intentional for tool execution	**F-03 Dynamic Function Resolution (GetProcAddress + GetModuleHandleA)** The VBA resolves API function addresses at runtime using GetModuleHandleA and GetProcAddress. This bypasses static import analysis and is a common technique used to avoid detection — legitimate extraction tools do not require this pattern.	
Intentional for tool execution	**F-04 VBA Project Password Bypass Routine** Module2 contains an 'unprotected()' subroutine that displays a message confirming the VBA project protection has been removed. This suggests the tool includes a self-unprotection mechanism, raising concerns about what protected code may be concealed.	
Intentional for tool execution	**F-05 DialogBoxParamA (user32) Custom Dialog** A custom Win32 dialog is invoked via DialogBoxParamA. While this may be used for credential input UI, its combination with the other findings warrants review. Legitimate tools typically use native Excel InputBox calls.	

RISK ASSESSMENT MATRIX

Ref	Risk Description	Likelihood	Impact	Rating	Mitigated?
R-01	Shellcode/malware execution via VirtualProtect + MoveMemory pattern	Medium	Critical	HIGH	Requires source code review
R-02	VBA project was or is self-unprotectable, hiding code from review	Confirmed	High	HIGH	Require unprotected source
R-03	Credential harvesting via custom Win32 dialog (DialogBoxParamA)	Low	High	MEDIUM	Review UI implementation
R-04	Dynamic API resolution evades static AV scanning	Medium	Medium	MEDIUM	Sandbox + behaviour analysis
R-05	Macro-enabled file executed outside of VDI / controlled environment	Medium	High	MEDIUM	Policy control (deployment)
R-06	Excessive SAP permissions granted beyond read-only roles	Low	High	MEDIUM	Role enforcement (SAP Basis)
R-07	Data exfiltration of extracted SAP interface data	Low	Medium	LOW	Network egress controls

RECOMMENDED CONTROLS & MITIGATIONS(ALREADY VALIDATED BY TARENTO)

01

RECOMMENDED

Obtain and review unprotected VBA source code

Request Tarento to provide the full unprotected VBA source code for Module1 and Module2 for code review. Do not approve the tool until the purpose of VirtualProtect, MoveMemory, PAGE_EXECUTE_READWRITE and GetProcAddress calls is independently verified.

02

RECOMMENDED

Detonate in sandboxed environment first

Execute the tool in an isolated sandbox with network monitoring (e.g. Defender ATP sandbox, Cuckoo, or Any.Run) to capture all API calls, network connections, and file system activity before allowing execution in any production or VDI environment.

03

RECOMMENDED

Submit to AV / EDR with behavioural scanning

Submit Fix.xlsm to enterprise AV/EDR for both static signature and dynamic behavioural analysis. Cross-check against VirusTotal. The PAGE_EXECUTE_READWRITE pattern is detectable by most modern EDR engines.

04

RECOMMENDED

Restrict execution to isolated VDI session

If approved, execution must be constrained to a dedicated VDI session with no access to production systems other than the target SAP PO environment. VDI sessions should be monitored and recorded.

05

RECOMMENDED

Enforce read-only SAP roles via SAP Basis

Confirm with SAP Basis that the service account used has only SAP_XI_API_DISPLAY_J2EE and SAP_XI_API_DEVELOP_J2EE roles assigned. Conduct a role audit before and after the discovery exercise.

06

RECOMMENDED

Network egress monitoring during execution

Monitor all outbound network connections from the host during tool execution. All traffic should be directed exclusively to the SAP PO server hostname/IP on the expected port. Any other connections should be treated as an incident.