Tidal Accelerator

Solution Deployment

Table of Contents

Table of Contents

Deployment
- Overview
- Implementation

Network Connectivity
- Outbound connectivity
- On-Premise internal connectivity
  - Infrastructure Data
    - Windows Servers
    - Unix Servers
  - Database Analysis
  - Web Application Analysis
  - vSphere Integration

Authorization
- Source Code Analysis
- Database Analysis
- Infrastructure Data - vSphere Integration
- Infrastructure Data - Machine Stats
  - Windows servers
  - Unix servers

Data collected
  - Source Code Analysis
  - Database Analysis
  - Infrastructure Data

Appendix

Deployment

Overview

The Tidal Migrations discovery process gathers and assesses data from several distinct sources. These sources include server (physical or virtual) infrastructure, database engines (Oracle, SQLServer, etc.), vSphere, application source code files, DNS configuration, and web-based application and services network endpoints. For each of these sources, access requirements are kept to what is minimally required. If certain sources are not available or applicable to your project, they are not required.

Implementation

There are several options available in deploying Tidal Migrations into your environment. It is best to choose the one that is simplest and easiest for you, and that meets all of our own security and operational requirements.

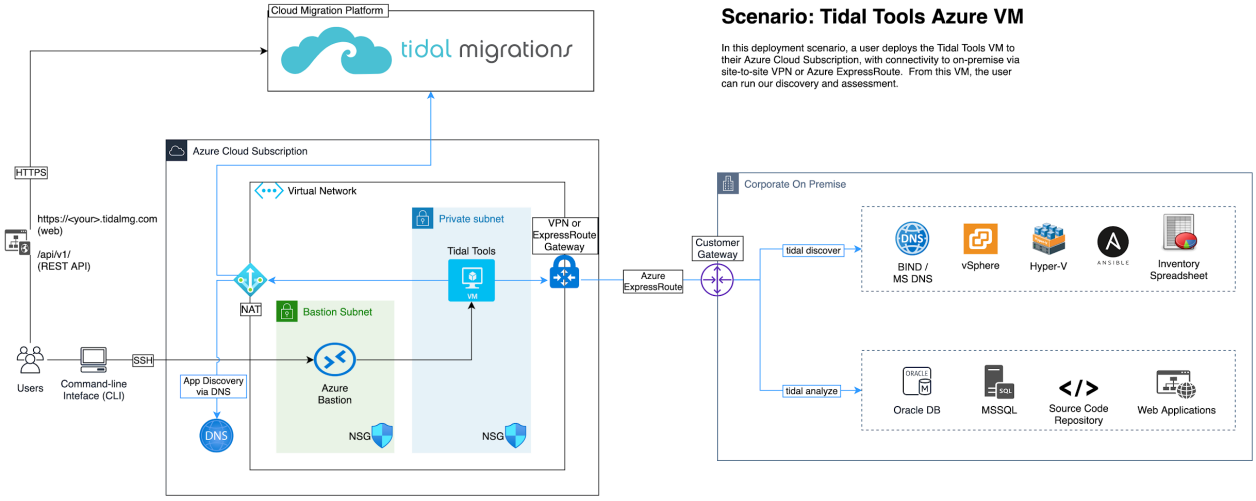
Depending on the deployment pattern you choose to use, you can obtain the software from several sources:

- AWS Marketplace - EC2 based
    - This allows you to deploy an EC2 instance in your AWS account, and control and operate the discovery process to your source environment from there.
- Azure Marketplace - Virtual Machine based

    - This allows you to deploy an EC2 instance in your AWS account, and control and operate the discovery process to your source environment from there.
- GCP Marketplace - Virtual Machine
    - This allows you to deploy a virtual machine in your project, and control and operate the discovery process to your source environment from there.
- Tidal Migrations Appliance - OVA-based deployment
    - This allows you to deploy a virtual machine in a virtual environment of your choice, and control and operate the discovery process to your source environment from there.
- Install on any device
    - This allows you to install the needed software on any device running Linux, Windows, or macOS. It can be a personal laptop, physical, or virtual server, and you can control and operate the discovery process to your source environment from there.
    - The required software installation includes:
        - Tidal Tools
        - Machine-Stats
        - Docker
        - Optionally - DNS Tools or Nmap

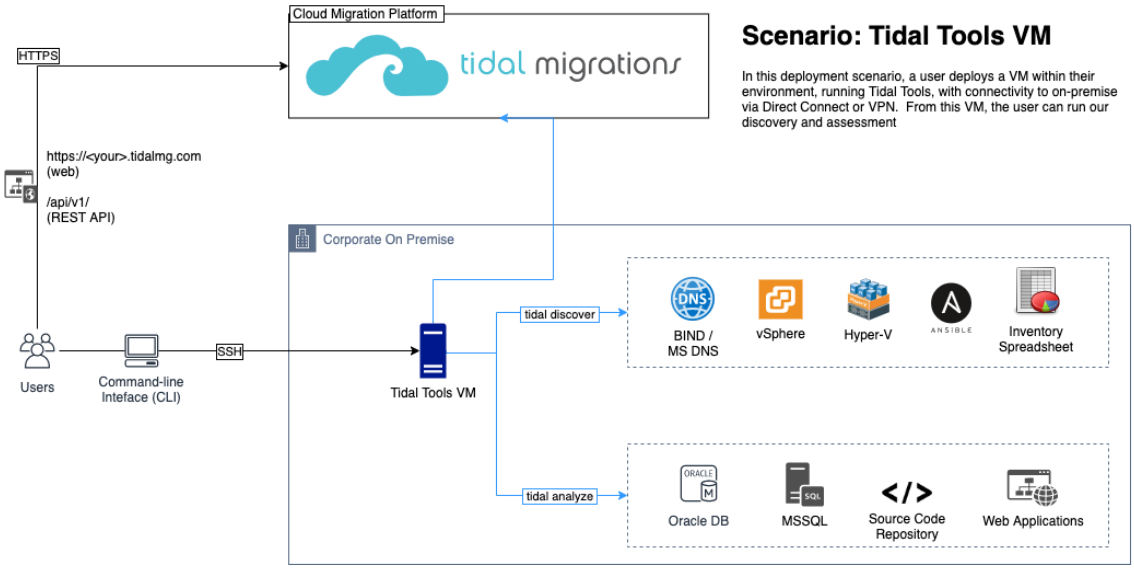
Scenario A - Tidal Tools Azure VM

Deploy to an Azure Subscription, with connectivity to the on-premise environment being migrated.



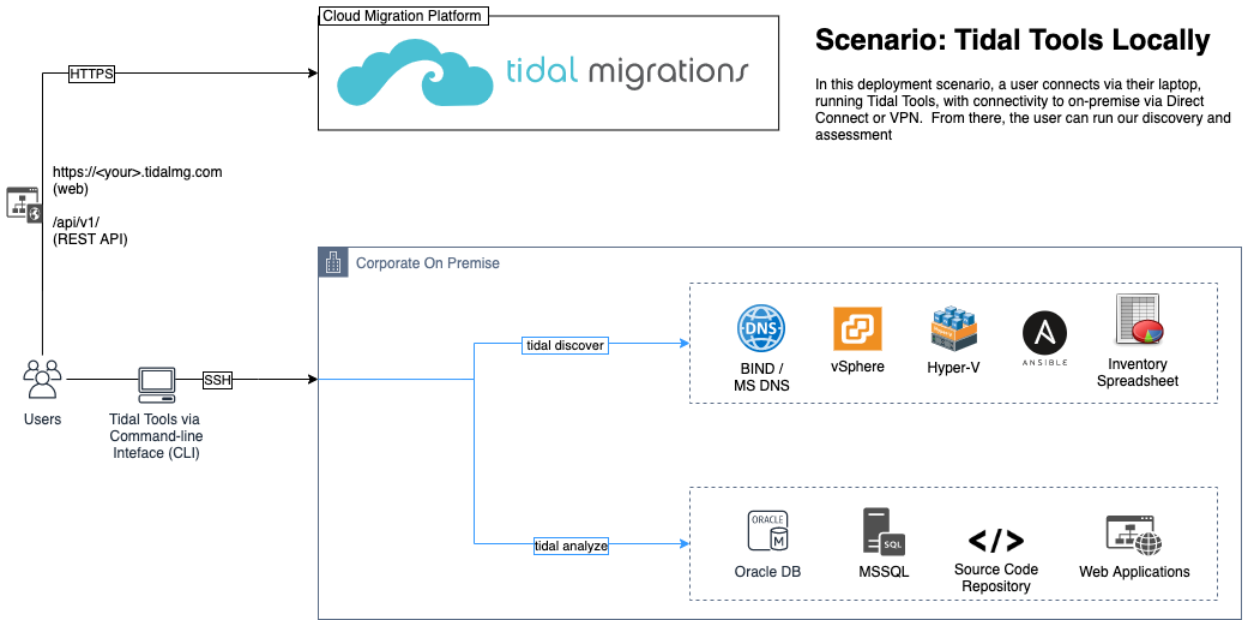
Scenario B - Tidal Tools VM

Deploy to a virtual machine within the on-premises environment being migrated.



Scenario C - Tidal Tools Locally

Install needed tools on any device and connect to the on-premises environment being migrated.



Other implementations are also possible such that the required connectivity is established.

Network Connectivity

Outbound connectivity

Outbound connectivity is required to get the data from your source environment to the Tidal Migrations platform.
This can be done in multiple ways, either directly or indirectly. If direct, then outbound connectivity is required from the location running Tidal Migrations software in your environment. If indirect, data must be transferred from the source to another location that has outbound connectivity.

The data must be transmitted to the Tidal Migrations platform via port 443 over HTTPS.

HTTPS (port 443) access to *.tidalmg.com (If not wildcard access, access to app.tidalmg.com and your_workspace_name.tidalmg.com is needed.)

To run source code analysis and database analysis, access to gcr.io is needed to download docker images used for the analysis.

HTTPS (port 443) access to gcr.io

On-Premise internal connectivity

Infrastructure Data

Machine Stats, Tidal Migrations' discovery tool, requires connectivity to any servers (virtual or physical) in scope for the project to capture telemetry data from the infrastructure.

Windows Servers

For Windows servers, there are two connectivity options available.

WMI method

This option uses a dynamic range of ports, via Windows DCOM.

- **Port 135 opens inbound** to the machine where Tidal Migrations is running.
- **Ports 1024 - 65535 open outbound** by default, from any of the in-scope Windows servers for which telemetry is being collected. Alternatively, WMI can be configured to explicitly use a fixed port. See this Microsoft documentation for how to configure that. In such cases, only that specific port is needed.

winRM method

This option requires **ports 5985 and 5986** to be open, between the target servers and the server running Machine Stats.

Unix Servers

Port 22 is needed to establish SSH connectivity. If SSH is on a non-default port, access is needed on the port it is running on.

Database Analysis

Network connectivity to all the databases in scope should be established via the port on which the databases accept incoming connections. The default ports for common databases include:

Oracle	**1521**
SQLServer	**1433**
Postgres	**5432**
MySQL	**3306**

Web Application Analysis

This discovery and analysis require connectivity on all ports that are used to serve HTTP/HTTPS web applications and services. The **defaults are ports 80 and 443**.

vSphere Integration

This integration requires connectivity to any vSphere VCenter servers, where the VCenter API is served from.
The default is port 443.

Authorization

Source Code Analysis

Read access for the application source code files is required to perform the analysis. The source code files can be copied, and the analysis run, from any

machine. Source code analysis does not require any network or connectivity access to any running environments.

Database Analysis

Tidal Migrations' database analysis is based on certain roles granted to the database user who can access and capture certain metadata from the databases. No access to the data held in the databases is required. The specific database user roles that are required for database analyses are defined in the script detailed on this link.

Infrastructure Data - vSphere Integration

A read-only user account with access to vCenter. See the guides for more information.

Infrastructure Data - Machine Stats

The code executed against all the servers is open source and can be reviewed on GitHub.

Windows servers

To collect the data from Windows servers a Windows user or domain account with access to execute unsigned Powershell scripts on any of the in-scope Windows servers is required.

Note, that Machine Stats is deliberately unsigned to allow the code to be inspected, customized, or modified as required.

Unix servers

Requires an OS (Linux, AIX, etc.) user account to authenticate to the server, either via a private key or password. User access requires the ability to invoke Python (version 2.6 or newer) on all of the target servers.

Data collected

Source Code Analysis

The entire code analysis process takes place locally on your machine. It will scan your files locally looking for common patterns and information when modernizing. This information includes:

- Statistics of total number of lines, by language. For example:

JavaScript


```
{"Blank": 575,
 "Bytes": 95722,
 "Code": 2530,
 "CodeBytes": 0,
  "Comment": 299,
  "Complexity": 183,
  "Count": 42,
  "Files": [],
   "Lines": 3404,
   "Name": "JavaScript",
   "WeightedComplexity": 0}
```

- Filepaths of files analyzed, for example:

```
JavaScript

"/src/home/pnr/tidal/schoolbus/Server/SchoolBusAPI/Authentication/SbJwtBearerEvents.cs",
"/src/home/pnr/tidal/schoolbus/Server/SchoolBusAPI/Authorization/ClaimsPrincipalExtensions.cs",
"/src/home/pnr/tidal/schoolbus/Server/SchoolBusAPI/Authorization/PermissionHandler.cs"
```

- Identified software dependencies package names, and versions, for example:

```
JavaScript
{"from_file":
"/src/home/pnr/tidal/schoolbus/Server/Serilog.Sinks.SbiPostgreSql/Serilog.Sinks.SbiPostgreSql.csproj",
"language": "CSharp",
"name": ".NET Core",
"owner": null,
"version": "3.1"
},
{"from_file": "/src/home/pnr/tidal/schoolbus/client/package-lock.json",
"language": "JavaScript",
"name": "@babel/code-frame",
"owner": null,
"version": "7.10.4"
}
```

- Identified patterns and implementations in code, for example, the detection of filesystem usage in Python:

JavaScript


```
{"check_id": "filesystem-python",
 "end": {
 "col": 19,
 "line": 132,
 "offset": 3703},
 "extra": {
 "engine_kind": "OSS",
 "fingerprint":
"f50c74db097532ad463e76e178545defc5ed47fbaf364527da876a75e33481708f97308217f236
4b57a9986f939c65030a21853622be9380d16c08940a95cbb6_0",
 "is_ignored": false,
 "lines": "    if not os.path.exists(output_dir):",
 "message": "Filesystem usage",
 "metadata": {},
 "metavars": {},
 "severity": "INFO"},
 "path": "/src/home/pnr/tidal/schoolbus/ApiSpec/CCWJurisdiction/csv2json.py",
 "start": {
 "col": 12,
 "line": 132,
 "offset": 3696}
}
```


These are snippets from a real analysis results file. To verify and confirm these findings, and see what the full results include on either your application or a sample application see these steps.

Database Analysis

Tidal Migrations' database analysis captures details about database implementation, data types, database features, column types, views, functions, application connection history, etc. Database metrics, including the following, are also collected:

- Server architecture
- CPU count
- Storage used
- RAM allocated
- Maximum storage requests
- Maximum storage throughput
- SGA size
- Maximum number of sessions
- Number of sessions used

User and application data within databases is not queried.

Infrastructure Data

Several data points are collected from either Machine Stats or the vSphere integration:

- IP addresses
- Internal URLs
- DNS records
- Technology names and versions identified
- Disk storage usage and allocations
- CPU type
- CPU count

- Hostname
- FQDN
- Operating system names and versions
- RAM Allocated (GB)
- RAM Used (GB)
- Total Virtual Memory
- Total Visible Memory
- Storage Allocated (GB)
- Storage Used (GB)
- CPU Count
- CPU Name
- CPU Description
- CPU L2 Cache Size
- CPU L3 Cache Size
- CPU Utilization
- CPU Socket Designation

These data points may be collected as a single point in time observation or as a time series (up to 30 days) depending on customer requirements.

AWS SCT

AWS SCT getting started guide

Appendix

User documentation for installation and discovery process
guides.tidalmg.com

Machine Stats source code and installation and documentation instructions
github.com/tidalmigrations/machine_stats

Tidal Migrations Security Information
tidalmigrations.com/security