Privacy DAM Manager guide

1. Product Overview

1.1. Introduction: What is a Privacy DAM?

As the personal data controller's responsibility for the protection of personal data becomes more important, the need for a way for the data controller to view and control personal data processing activities is increasing.

Privacy DAM makes the accesses to the database, which is the storage of personal information, into APIs, manages the processing type, processing purpose, and authority for individual APIs, and provides the status and analysis report on the results of individual API calls. Via Privacy DAM, the DPO(Data Protection Officer), a person in charge of personal information protection, can understand and respond to information processing activities with ease.

- Control of accesses to personal information storage (DB) through APIs
- Inspect and control the purpose and role of data processing per personal data handlers.
- Provides a dashboard for API call status and analyzed reports.

This Privacy DAM Manager is a part of the Privacy DAM solution. The Privacy DAM solution consists of a Privacy DAM Manager and Privacy DAM API Processors. A Privacy DAM Manager controls the data accessing APIs including a creation of new data accessing APIs and an activation/deactivation of data accessing APIs. It also manages who can call APIs and on what purpose the API can be called. The Privacy DAM API Processors are the actual worker processes that perform the SQL process according to the defined via a Privacy DAM Manager. The Privacy DAM API Processors are not included in this AMI. If you are interested in it, please contact the TOVDATA sales representatives. (contact@tovdata.com)

1.2. Prerequisites and Requirements

- Prerequisites

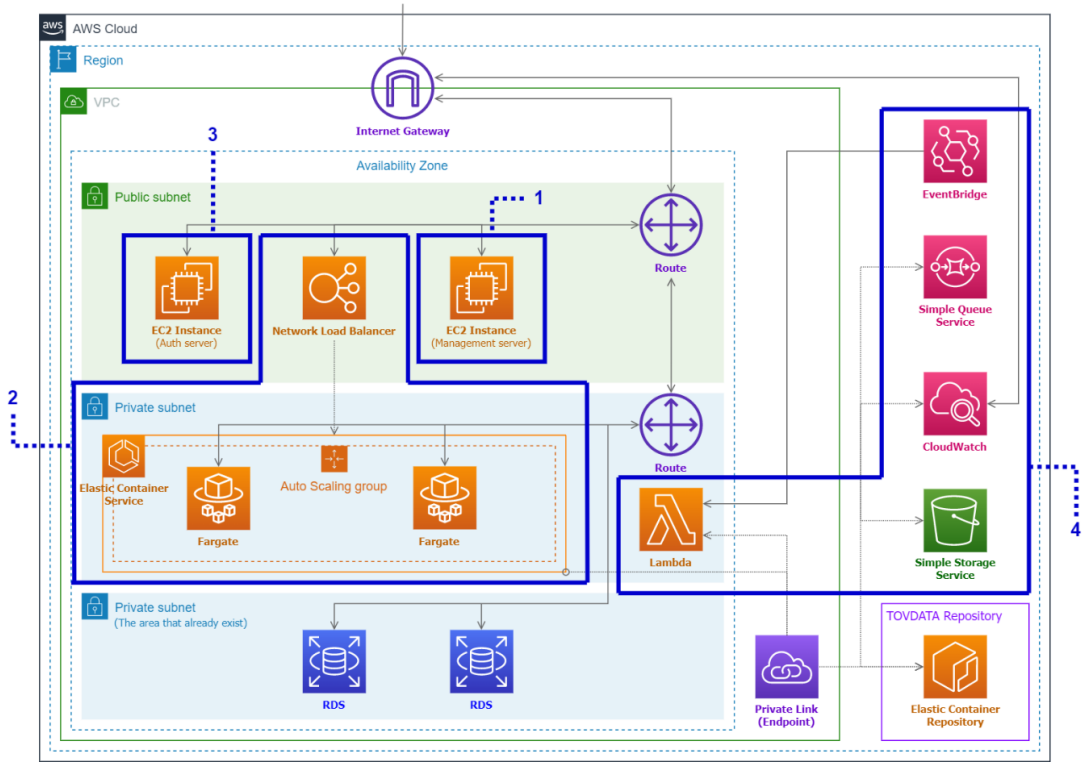
  The Privacy DAM Manager AMI is completely self-contained. You don't need to install any additional software to run and evaluate the Privacy DAM Manager. Basic AWS skills related with creating and managing EC2 instances are sufficient to deploy Privacy DAM Manager on AWS. In addition, basic Linux CLI familiarity

are preferred during the initial deployment for running some CLI commands.

- **Requirement**
  All you need are a publicly accessible subnet and a permission to launch an EC2 instance in that subnet.
- **Region support**
  There is no technical restriction to deploy and run the Privacy DAM Manager AMI.

1.3. Architecture Diagrams

The overall solution architecture of the Privacy DAM is shown in the following figure. Though the Privacy DAM Processor modules are not included in this AMI but those are shown in the figure for your information.



1. Privacy DAM Manager

- Located in a public subnet.
- Manages the DB access API information and provides them to the Privacy DAM API Processor.
- Provides a monitoring dashboad for the Privacy DAM API Processor status and API call log reports

2. Privacy DAM API Processor (Worker module)

- Processes the actual DB access and additional data processing according to the registered API information.

3. Privacy DAM API Processor (Auth module)

- Checks whether the API caller has right permisson and suitable purpose to call the API.

4. Privacy DAM API Processor (LogProcess module)

- Collecting the API call logs and user authentication logs, performs analysis to provide monitoring information for dashboard and archives log data and reports.

1.4. Security

Inbound traffic to port 22 must be allowed because it is required to access the instance using ssh. Also, inbound traffic to port 80 (HTTP) must be allowed to access the Privacy DAM Manager web pages.

In addition, inbound traffic for port 5432 must be allowed for Privacy DAM API Processor to access internal database from processing module and authentication module, and inbound traffic for 9091 must be allowed to measure the performance of the processing module.

- Allow inbound traffic anywhere: 80
- Allow inbound traffic for server manager: 22
- Allow inbound traffic for private subnet: 5432, 9091

1.5. Cost

We do not charge for downloading and testing this Privacy DAM Manager.

Only the cost of operating the AWS EC2 instance will be charged from AWS. The cost of operating an instance depends on the type of instance. For more information, visit the AWS site.

1.6. Size

- Instance type

  The minimum instance type for the Privacy DAM Manager is a general purpose EC2 instance type of t2.small or higher performance types. (More than 2 vCPUs and 4GiB memory are recommended for the initial deployment and operations.)

- Storage

  The minimum required EBS storage size for the Privacy DAM Manager is 8GB or larger.

2. Initial Deployment (an Amazon EC2 instance)

2.1. Create and set an instance for deploy

- Launch an EC2 instance onto the designated public subnet with the shared Privacy DAM AMI.
- Make sure that the public subnet does not have any restriction of connecting from the Internet to the instance via TCP ports 22 and 80. (If you are not sure of this, check the network ACLs of the public subnet.)
- Set a security group that allows the TCP connections to the port number 22 (for SSH) and to the port number 80 (for HTTP) from the Internet (i.e., 0.0.0.0/0) or from your corporate network IP. Make sure a Public IP address is auto-assigned (i.e., Enable 'Auto-assign Public IP')
- Create and attach the EBS volume with the instance.
- After a successful deployment of an EC2 instance resource, continue the deployment stage of the Privacy DAM Manager inside the instance.
- Connect to the Privacy DAM Manager instance via SSH.

2.2. Configuration

Configure the Privacy DAM Manager passwords to access.

Command) privacydam config

```
ubuntu@ip-172-31-8-229:~$ privacydam config
Input an auth code to access management page: abcd1234
Setting completed
```

2.3. Run an application

Before you can run an application, you must run the database. Use the command below to run the database.

Command) sudo systemctl start postgresql

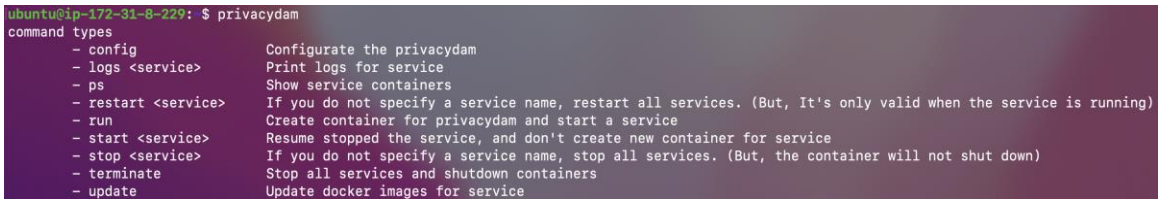
Then, type below command to create containers for application and run the application.

Command) privacydam run

```
ubuntu@ip-172-31-8-229:~$ privacydam run
[+] Running 4/4
 ✔ Network privacydam-mgm_privacydam  Created
 ✔ Container privacydam-back          Started
 ✔ Container privacydam-front         Started
 ✔ Container privacydam-proxy         Started
```

Now, the initialization is done. You can now connect the Privacy DAM manager web interface using your favorite browser.

2.4. Additional commands for operations

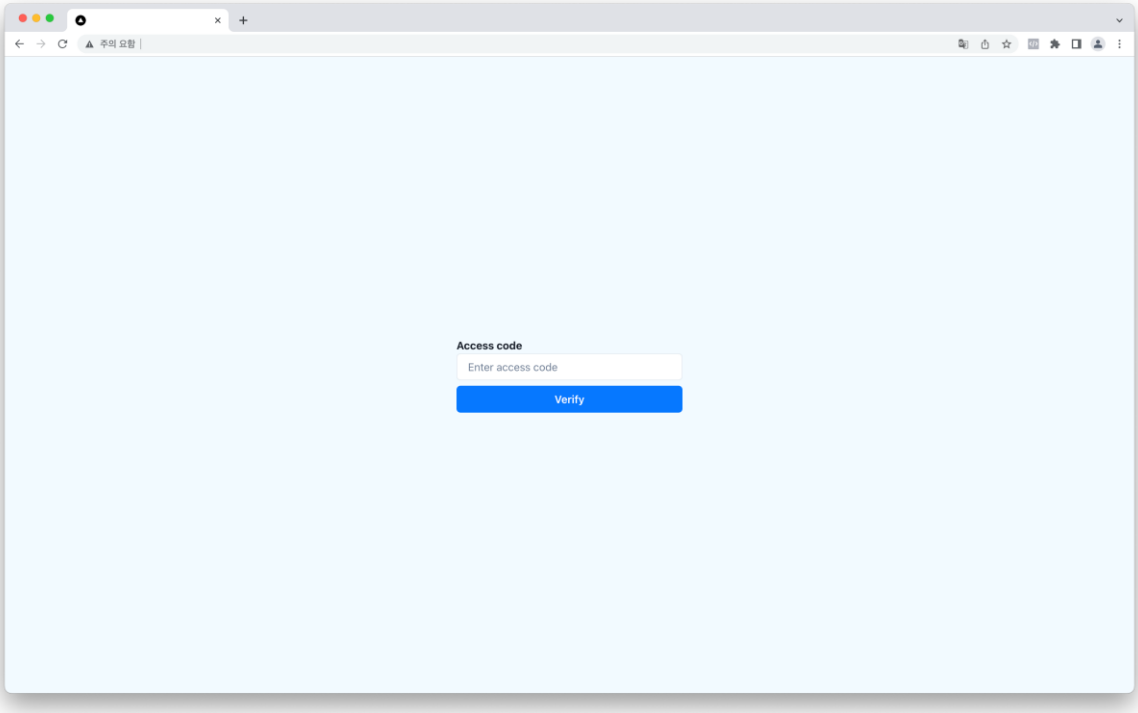


There are some additional commands that might be useful for your operations. The application consists of services named proxy, frontend, backend.

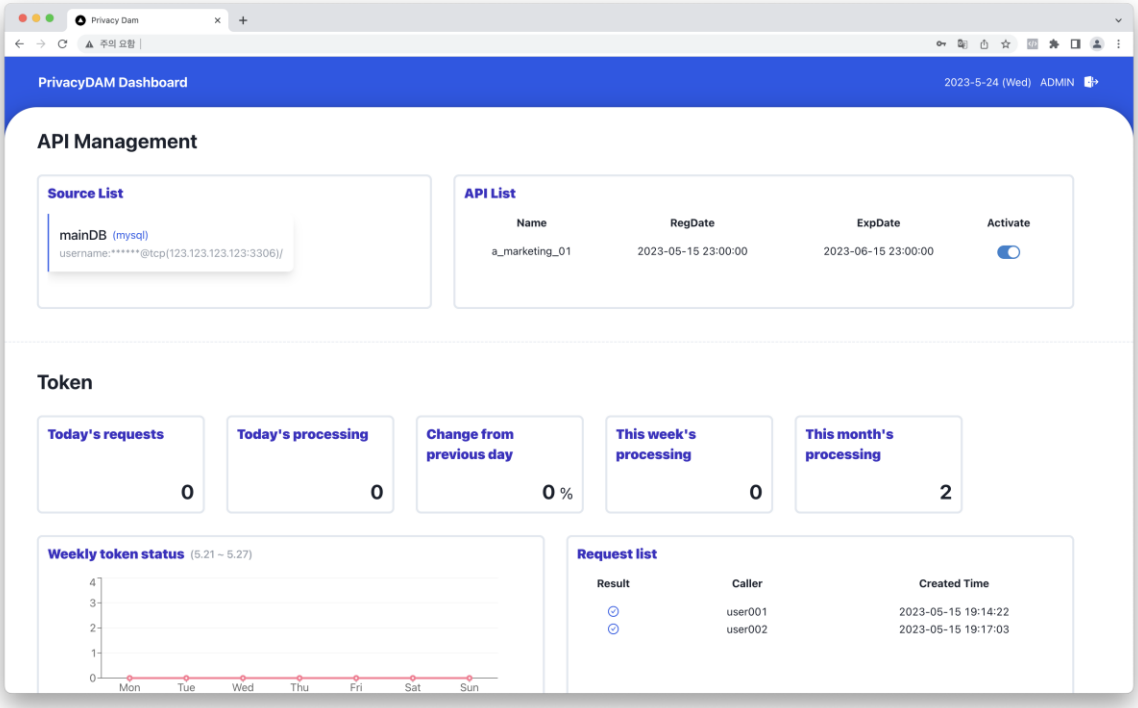
- Command) privacydam logs <service_name>
  Display the real-time log of the service. If you don't specify a service, displays all service logs for the application.
- Command) privacydam ps
  Display a list of currently running service.
- Command) privacydam restart <service_name>
  Restart the service. If you do not specify a service, restart all services.
- Command) privacydam run
  Create containers for application and run the application.
- Command) privacydam start <service_name>
  Run the service. If you don't specify a service, it runs all services within the application. The "privacydam run" command must be preceded and containers must be existed for the application.
- Command) privacydam stop <service_name>
  Stop the service and keep the container for retention of the data but containers and its data will not be deleted. If you don't specify a service, stop all services within the application.
- Command) privacydam terminate
  Stop all services within the application and delete all containers for application.
- Command) privacydam update
  Proceed with the update for all services within the application only if there is an update.

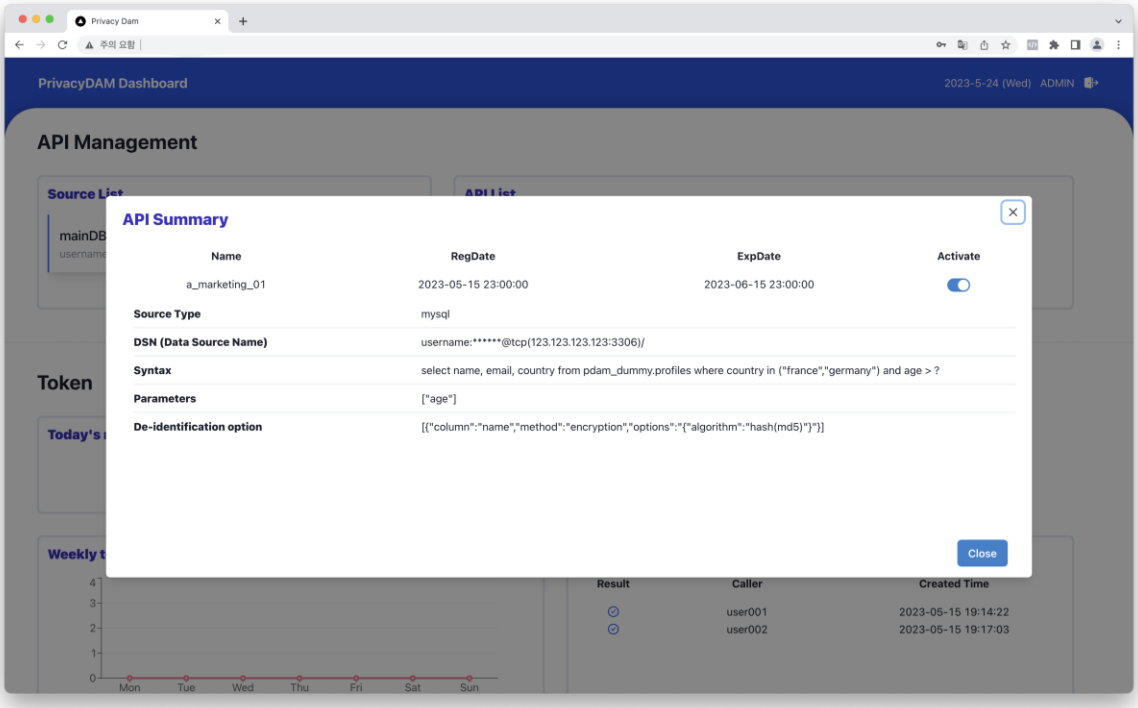
3. Screenshots

- Initial page of Privacy DAM (Enter the access code as you set during the CLI setup sessions)



- Main dashboard page of Privacy DAM




Privacy Dam
주의 요함
PrivacyDAM Dashboard
2023-5-24 (Wed) ADMIN
API Management
Source List
mainDB
API List
Token
Today's
Weekly
API Summary
Name
RegDate
ExpDate
Activate
a_marketing_01
2023-05-15 23:00:00
2023-06-15 23:00:00
Source Type
mysql
DSN (Data Source Name)
username:******@tcp(123.123.123.123:3306)/
Syntax
select name, email, country from pdam_dummy.profiles where country in ("france","germany") and age > ?
Parameters
["age"]
De-identification option
[{"column":"name","method":"encryption","options":"{"algorithm":"hash(md5)"}"}]
Close
Result
Caller
Created Time
user001
2023-05-15 19:14:22
user002
2023-05-15 19:17:03
4
3
2
1
0
Mon
Tue
Wed
Thu
Fri
Sat
Sun